

University of Mississippi

eGrove

Electronic Theses and Dissertations

Graduate School

2017

Deployment, Coverage And Network Optimization In Wireless Video Sensor Networks For 3D Indoor Monitoring

Tisha Lafaye Brown
University of Mississippi

Follow this and additional works at: <https://egrove.olemiss.edu/etd>



Part of the [Computer Sciences Commons](#)

Recommended Citation

Brown, Tisha Lafaye, "Deployment, Coverage And Network Optimization In Wireless Video Sensor Networks For 3D Indoor Monitoring" (2017). *Electronic Theses and Dissertations*. 953.
<https://egrove.olemiss.edu/etd/953>

This Dissertation is brought to you for free and open access by the Graduate School at eGrove. It has been accepted for inclusion in Electronic Theses and Dissertations by an authorized administrator of eGrove. For more information, please contact egrove@olemiss.edu.

DEPLOYMENT, COVERAGE AND NETWORK OPTIMIZATION IN WIRELESS
VIDEO SENSOR NETWORKS FOR 3D INDOOR MONITORING

A Dissertation
presented in partial fulfillment of requirements
for the degree of Doctor of Philosophy
in the Department of Computer and Information Science
The University of Mississippi

by
Tisha L. Brown
August 2017

Copyright Tisha L. Brown 2017
ALL RIGHTS RESERVED

ABSTRACT

As a result of extensive research over the past decade or so, Wireless Sensor Networks (WSNs) have evolved into a well established technology for industry, environmental and medical applications. However, traditional WSNs employ such sensors as thermal or photo light resistors that are often modeled with simple omni-directional sensing ranges, which focus only on scalar data within the sensing environment. In contrast, the sensing range of a wireless video sensor is directional and capable of providing more detailed video information about the sensing field. Additionally, with the introduction of modern features in non-fixed focus cameras such as the Pan, Tilt and Zoom (PTZ), the sensing range of a video sensor can be further regarded as a fan-shape in 2D and pyramid-shape in 3D. Such uniqueness attributed to wireless video sensors and the challenges associated with deployment restrictions of indoor monitoring make the traditional sensor coverage, deployment and networked solutions in 2D sensing model environments for WSNs ineffective and inapplicable in solving the Wireless Video Sensor Network (WVSN) issues for 3D indoor space, thus calling for novel solutions.

In this dissertation, we propose optimization techniques and develop solutions that will address the coverage, deployment and network issues associated within Wireless Video Sensor Networks for a 3D indoor environment. We first model the general problem in a continuous 3D space to minimize the total number of required video sensors to monitor a given 3D indoor region. We then convert it into a discrete version problem by incorporating 3D grids, which can achieve arbitrary approximation precision by adjusting the grid granularity. Due in part to the uniqueness of the visual sensor directional sensing range, we propose to exploit the directional feature to determine the optimal angular-coverage of each deployed visual sensor. Thus, we propose to deploy the visual sensors from divergent directional angles

and further extend k-coverage to “k-angular-coverage”, while ensuring connectivity within the network. We then propose a series of mechanisms to handle obstacles in the 3D environment. We develop efficient greedy heuristic solutions that integrate all these aforementioned considerations one by one and can yield high quality results. Based on this, we also propose enhanced Depth First Search (DFS) algorithms that can not only further improve the solution quality, but also return optimal results if given enough time. Our extensive simulations demonstrate the superiority of both our greedy heuristic and enhanced DFS solutions. Finally, this dissertation discusses some future research directions such as in-network traffic routing and scheduling issues.

DEDICATION

*To my husband, De'Andre Gaines and my parents Mr. Lorenzo and Pecola Brown,
without your continuous support and respect for education this would not have been possible.
To my sister and niece, I live to inspire you.*

LIST OF ABBREVIATIONS

AA	Air-to-Air
AS	Air-to-Sand
SA	Sand-to-Air
SC	Simple Cubic
SS	Sand-to-Sand
HS	High Resolution
ABC	Artificial Bee Colony
ACO	Ant Colony Optimization
PSO	Particle Swarm Optimization
WIFI	IEEE 802.11x
NCA	Network Coverage Area
DCQ	Deployment Coverage Quality
MMAS	Max-Min Ant System
NRSA	Network Relevant Sensing Area
DKC	Directional k-Coverage
SNR	Signal to Noise Ratio
RSSI	Received Signal Strength Indicator
CBR	Constant Bit Rate
TES	Transform Expand Sample
ID	Identically Distributed
VSN	Visual Sensor Network

ROI Region of Interest

UWSN Wireless Underwater Sensor Network

WMSN Wireless Mobile Sensor Network

WUSN Wireless Underground Sensor Network

WTSN Wireless Terrestrial Sensor Network

MI Magnetoinductive waveguide

GCD Greatest Common Divisor

LCM Least Common Multiple

WSN Wireless Sensor Network

WVSN Wireless Video Sensor Network

ARPANET Advanced Research Projects Agency Network

CCD Charge Coupled Device

MWSN Multimedia Wireless Sensor Network

CCTV Closed Circuit Television

IP Internet Protocol

QoS Quality of Service

CBR Constant Bit Rate

TES Transform Expand Sample

IID Independently and Identically Distributed

PTZA Pan Tilt Zoom Algorithm

FoV Field of View

WMN Wireless Mesh Network

WLAN Wireless Local Area Network

ORPD Optimal Regular Pattern Deployment

ACKNOWLEDGEMENTS

First and foremost, I would like to give honor and praise to my Lord and Savior Jesus Christ, who has provided me with all of my Earthly needs and allowed me to fulfill this incredible goal of mine, in attaining a PH.D.

To my loving and supportive family, Lorenzo and Pecola Brown, my sister Latoya, aunt Glenda Porter, grandparents Willie P. Brooks, Estella O'Neil, John and Nell Brown and niece Bryanna, I love you. Mom you are the real MVP. You all are greatly appreciated.

Professionally, I would like to thank my advisor Dr. Feng Wang for guiding me with your immense knowledge and directing me to pursue high-quality research, giving me the courage to explore my own ideas and helping me develop them. I am forever grateful. Your guidance over the last four years have been an invaluable asset to me and have lead me to reaching my goal of finishing my degree. Special thanks to Zhonghui (John) Wang for being a great colleague, research collaborator and friend. John, your depth of knowledge and insightful thoughtfulness helped me in developing my research skill set. I am also thankful to all of the current and former Future Networks Research group members: Tong Shan, Zhonghui (John) Wang, Khaled Ahmad Saleh Sabahein and Ghebrekristos Haileslassie for their help and invaluable discussion and comments. I would also like to thank Dr. Jianxia Xue for providing insightful assistance in the development of the simulator.

Special thanks are extended to the Computer and Information Science Department, especially Dr. Dawn Wilkins and Mrs. Carrie Long. I would also like to thank my committee members: Dr. Dawn Wilkins, Dr. Byunghyun Jang and Dr. Donald Cole, thank you so much for agreeing to help me through this process. Additionally, I would like to thank all the faculty members of the Computer and Information Science department, colleagues and associates that have played a significant role in me getting to this stage in my academic career

including my Mississippi Valley and Jackson State University family. To the SREB staff, thanks for your financial support and academic guidance over the years. I am so honored to be an alumni of the University of Mississippi, it was a long and hard process but well worth the reward.

Last but definitely not least, to my incredible and supportive husband, DeAndre Gaines, you are my rock and I love you so much. May God continually bless and keep us strong and together forever.

TABLE OF CONTENTS

ABSTRACT	ii
DEDICATION	iv
LIST OF ABBREVIATIONS	v
ACKNOWLEDGEMENTS	vii
LIST OF TABLES	xiii
LIST OF FIGURES	xiv
INTRODUCTION	1
1.1 Research Challenges	3
1.2 Research Contributions	5
1.3 Dissertation Organization	7
BACKGROUND	9
2.1 Wireless Sensor Networks	9
2.2 Evolution of Wireless Sensor Networks	11
2.3 Multimedia Wireless Sensor Networks	14
2.4 Wireless Video Sensor Networks	16
2.4.1 History of WVSNs	18
2.4.2 Issues in WVSNs	19
2.4.3 Strategies	22

RELATED WORK	27
3.1 Wireless Sensor Networks	27
3.1.1 Deployment	28
3.1.2 Coverage	30
3.1.3 Connectivity	32
3.2 Wireless Video Sensor Networks	35
3.2.1 Deployment	36
3.2.2 Angular Coverage	38
3.2.3 Connectivity	39
3.2.4 Obstacle-Awareness	41
OPTIMAL DEPLOYMENT	44
4.1 Introduction	44
4.2 Model Comparison	45
4.3 Problem Formulation	49
4.3.1 Continuous Space Model	51
4.3.2 Discrete Space Model	52
4.4 Experimental Methodology	55
4.4.1 Greedy Heuristic Algorithm	55
4.4.2 Enhanced Depth First Search Algorithm	57
4.5 Experimental Results	60
CONNECTIVITY AND OBSTACLE-AWARENESS	66
5.1 Introduction	66
5.2 Model Comparison	68
5.3 Problem Formulation	69
5.3.1 Continuous Space Model	70
5.3.2 Discrete Space Model	72

5.3.3	Strategies for Obstacle Detection and Connectivity	73
5.4	Experimental Methodology	78
5.4.1	Greedy Heuristic	78
5.4.2	Enhanced Depth First Search	79
5.5	Experimental Results	82
	2-ANGULAR-COVERAGE	89
6.1	Introduction	89
6.2	Coverage Comparison	90
6.3	Problem Formulation	94
6.3.1	Continuous Space Model	94
6.3.2	Discrete Space Model	98
6.4	Experimental Methodology	98
6.4.1	Greedy Heuristic Algorithm	99
6.4.2	Enhanced Depth First Algorithm	99
6.5	Experimental Results	102
	CONCLUSION AND FUTURE WORK	110
7.1	Conclusion	110
7.2	Future Work	111
7.3	Research Publications	113
	BIBLIOGRAPHY	115
	APPENDICES	128
	Greedy Heuristic Algorithm Complexity Analysis	130
	Enhanced DFS Algorithm Complexity Analysis	132
	Code Listing: Obstacle Constraint	135

Code Listing: Connectivity Constraint	139
Code Listing: Angular Coverage Constraint	141
VITA	142

LIST OF TABLES

4.1	Comparison of 2D and 3D Models	49
4.2	Parameter Notation Chart	51
4.3	Deployment Problem: Simulation Parameters	62
5.1	Parameter Notation Chart II	71
5.2	Connectivity and Obstacle Awareness: Simulation Parameters	83
6.1	2-Angular Coverage: Simulation Parameters	103

LIST OF FIGURES

2.1	The Beginning of ARPANET [1]	11
2.2	Evolution of Sensor Cost [2]	12
2.3	Applications Market Gains [2]	13
2.4	Types of Wireless Sensor Networks	15
2.5	Traditional WSN sensing range models. (a) 2D directional sector coverage model; (b) 2D omnidirectional sensing range model	16
2.6	Components of a Wireless Video Sensor Node	17
2.7	Sensor Node Comparison [3]	18
2.8	Coverage Classifications (a) Target Coverage (b) Area Coverage (c) Barrier Coverage using pointillism of discrete points [4]	20
2.9	Connected Directed Digraphs (a) Strongly connected graph (b) Weakly connected graph	22
2.10	Common Deployment Strategies (a) Computational Geometry-based; as shown using Delaunay Triangulation (b) Grid-based (c) Force-based as shown using Virtual Force theory [5; 6]	24
4.1	2D Deployment Model	47
4.2	3D Sensor Deployment Environment	48
4.3	3D Environment	50
4.4	Facing direction sphere of video sensor with granularity of direction g_D	53
4.5	Video Sensor Parameters	54
4.6	3D Lattice Model	55
4.7	Network Simulation Environment	61
4.8	Dimensional 3D Indoor Space	63
4.9	Varying g_A impact	63
4.10	Varying granularity for g_D	63
4.11	Varying granularity for g_L	63
4.12	Impact of varying near fields	64
4.13	Varying field of views (FoVs)	64
4.14	Impact of varying far fields	65
4.15	Impact of candidate domain size	65
5.1	3D Lattice Grid Model	73
5.2	Representation of Lamp Obstacle (a) Actual Purchasable Lamp (b) Cuboid Lamp representation	74
5.3	Representation for Divide and Conquer Detection strategy	75
5.4	Accurate Detection Model	76
5.5	Simulated Indoor Obstacles	78
5.6	Variations of Indoor Space	84
5.7	Varying Obstacle Dimensions	84

5.8	Obstacle per Unit Length	85
5.9	Varying Connectivity Range	85
5.10	Impact of Near Field	86
5.11	Variation of Field of View	86
5.12	Varying g_A	87
5.13	Varying g_D	87
5.14	Varying Impact of C_L	87
5.15	Varying Impact of g_L	87
5.16	Simulated Deployment Scenario considering Single Obstacle	88
5.17	Simulated Deployment Scenario considering Multiple Obstacles	88
6.1	Coverage Model Comparison (a) Target Coverage (b) Area Coverage (c) Barrier Coverage using pointillism of discrete points [4]	91
6.2	Concept Model for 2-angular-coverage	97
6.3	Variation of Deployment Domain	104
6.4	Varying Field of View	104
6.5	Varying Near Field	105
6.6	Varying 3D Indoor Space	105
6.7	Impact of Varying g_L	106
6.8	Varying Impact of g_A	106
6.9	Impact on Varying the Far Field	107
6.10	Impact of Varying g_D	107
6.11	Running Time	108

CHAPTER 1

INTRODUCTION

Today's wireless sensor networks (WSNs) are a far cry from the earlier types of wired sensors, which have been transformed into state of the art devices that provide a new generation of features to offer to consumers [7]. Historically, WSNs were first established in the 1950s by the United States government to provide the military and other government agencies with a new form of advance technology which enabled surveillance of foreign entities and allowed for secure communication among our government [8]. After the success of WSNs in that role an effort began to explore the capabilities of WSNs, specifically how WSNs could be improved by: reducing the cost of the sensors, reducing the energy consumption of sensors and enabling deployment in commercial applications [9]. In some cases sensor costs have declined by as much as ten fold over the past decade. Following the initiative to solve the aforementioned challenges associated with the growing WSN technology, the propagation of the technology expanded to include a myriad of WSNs subcategories that included: mobile [10], underground [11], underwater[12], terrestrial [13] and multimedia WSNs [14].

Consequently, due in part to the growth of WSNs, economical cost and versatility of complementary metal-oxide-semiconductor (CMOS) image sensors, Wireless Video Sensor Networks (WVSNs) have emerged as a prominent technology capable of integrating into numerous applications including: industrial automation, home security, environmental monitoring and traffic surveillance [15]. This visual data and surveillance driven approach is currently at the forefront of industry innovations and research interests. Within the monitoring environment, the video sensor nodes are interconnected with each other over wireless connections which allow them to collaboratively accomplish tasks such as compression, retrieval and correlation of video data [16], to name a few. These factors make WVSNs very

different from traditional wireless sensor networks (WSNs), where scalar nodes such as thermal or light sensors are often considered in 2D and with omni-directional sensing range [17], i.e., usually a circular disc with a defined radius. On the other hand, WVSNs utilize video sensors that normally have a sensing range with a directional field of view (FoV), where the perspective is often modeled as a fan-shape in 2D and a cone-shape or a pyramid-shape in 3D [18]. Therefore, not only does the deployment location affect the area a video sensor covers but also the direction of the video sensor is important to its sensing ability. This challenge often renders the traditional deployment solutions for WSNs and 2D environments infeasible to be applied to solve the WVSN deployment problem [19] [20] [21].

Furthermore, there are also challenges directly coming from monitoring the indoor environment [22] [23]. For example, there are often obstacles such as ceiling lamps and furniture inside the indoor space, which, if not carefully considered, can easily block the line-of-sight of deployed video sensors and reduce their sensing capability. Another challenge is that when incorporating video sensors surveillance into a normal WIFI network, one has to take into account the large amount of data traffic collected by the video sensors. This may cause the WVSN to encounter interference or interruptions to the normal use of the WIFI communication within the building (e.g., an office area). Therefore, the WVSN is often required to have its own separate communication network to connect the entire monitoring system. On the other hand, although there are recent studies on the WVSN coverage and deployment problem, most of them were inapplicable to 3D indoor monitoring, either due to less practical assumptions such as video sensors that only face certain directions or still considering a 2D scenario, or because of simply negating the impact in which the angular direction of the video sensor affects the sensing capability, or failing to incorporate important issues such as obstacle awareness and network connectivity [24] [25] [26] [27] [28]. Moreover, we envision the use of this research in applications where monitoring is provided to cover spaces requiring strict specialized coverage constraints (e.g., target coverage in jewelry stores and museums to prevent theft and public areas to improve the quality of

face/identity detecting via area angular coverage) implemented to detect obstructions within the monitoring space, provide specialized facial recognition and determine sensor placement for optimal coverage in scenarios where human estimation is not accurate.

Therefore, this dissertation proposes optimization techniques and algorithmic solutions for the deployment problem of wireless video sensor networks in a 3-D indoor space. Herein, deployment is concerned with the placement and dispatching of sensors in a simulated 3D real-world environment. Beyond this general deployment problem, there are several issues that are considered as well, such as 1) incorporating connectivity which ensures that the entire network is fully connected using a separate wireless communication among the video sensor nodes to a base station, 2) implementing “2 -angular-coverage” where at least two video sensors monitor a region within the coverage area from different angles, and 3) demonstrating mechanisms to provide obstacle awareness within the monitored space. Furthermore, a discussion is given to identify future research directions including addressing traffic issues in WVSNs to maximize the throughput and minimize the delay of the traffic built up within the network. The subsection below summaries each issue to be addressed in our research and its significance.

1.1 Research Challenges

There are countless studies available that explore the deployment problem in WSNs [21] [22] [25]. However, there are major differences in WSNs and WVSNs, which prevent the use of techniques that are currently well developed for WSNs to be applied in WVSNs [26]. Introducing WVSNs into an environment presents additional challenges that are not often attributed to WSNs such as the quality of coverage in WVSNs that depend on the orientation of the video sensor. Another differentiating aspect of WVSNs versus that of WSNs is the sensing range of sensor nodes which is a function of the sensor’s field of view (FoV), aspect ratio and near/far fields. In this section, we will explore the research challenges in addressing the deployment, coverage and networked issues that arise in WVSNs for 3D

indoor monitoring.

One of the critical issues in WVSNs is the deployment problem. Dissimilar from traditional scalar sensors such as heat or light sensors that have omni-directional sensing ranges, the deployment of a visual sensor must consider the directional aspect of the sensor. This additional criterion can increase the cost associated with deploying sensors nodes. As an example, a scenario can happen where multiple nodes are deployed and cover the same area redundantly which increases the resource overhead, resulting in an inefficient deployment scheme. Additionally, deployment can affect the communication among nodes in the sensor network. Even though existing works explore the deployment problem in WSNs there is still a substantial need to address the issues for WVSNs in totality with concern given to the angular sensing field.

Another critical issue in WVSNs in contrast to WSNs is the support of high rate video streamed traffic. Due to the large amount of data that travels in WVSNs, a connectivity scheme to ensure communication among the nodes is desired. Connectivity in WVSNs for a monitored space is a necessity. For example, if a WVSN within a monitored space relies only on existing WIFI availability, there are circumstances that may occur where the system fails or become unavailable (i.e., power outage). In this scenario the bandwidth overhead increases and can interfere with the existing system (i.e., CS department WIFI) causing signal delays. Moreover, such consideration provides an advantage in developing separate communication platforms that are not reliant upon the existing connectivity systems to handle the impact of processing the large amount of data traffic collected by the video sensors.

In WVSNs a fault tolerant scheme is desired. Often times due to poor energy conservation (battery life) of wireless video sensor nodes within a network, a scenario can happen where one or more nodes can go offline resulting in the failure of a node to cover a specific region if only one node is responsible for that coverage area. There is thus a need to implement k-coverage in the network. k-coverage ensures that at least k sensors will be able to cover a specific region. Implementing k-coverage drastically improves the overall performance of

the WWSN. Often times existing studies primarily touch upon the deployment problem for 1-coverage, and only consider the 2D aspect of the sensing range coverage. If implemented as such, the deployment scheme would lack practical real world application. Also, in an indoor environment, deployment area restrictions further render the solutions for traditional scalar sensors and 2D sensing field incapable of solving the WWSNs aspect. Thus, it is more appropriate to consider k -coverage for WWSNs in 3D indoor space monitoring. To maximize the information that can be captured by visual sensors covering a 3D location, a scheme to deploy the visual sensors from different directional angles and further extend k -coverage to “ k -angular-coverage” to denote such unique requirement in WWSNs is befitting. In this dissertation, we take the first step to study the 2-angular-coverage.

Obstacle awareness in WWSNs is another challenging factor. In indoor spaces, obstacles such as decorative fittings, furniture and ceiling lights are staples of modern design in commercial and residential living spaces. However, these furnishings can easily block the line-of-sight of a video sensor. Works considering indoor 3D space coverage but failing to incorporate an obstacle-aware strategy for obstacles existing in the monitored space can produce results that are not feasible in real world scenarios.

1.2 Research Contributions

In this dissertation, we thoroughly investigated the aforementioned challenges associated within the general deployment problem of WWSNs that are not readily addressed in numerous well defined solutions for WSNs. Different from previous studies, we proposed solutions for the deployment problem in WWSNs for 3D indoor spaces with the consideration to ensuring coverage, connectivity, obstacle-awareness and reliability. Our work derives a more precise network model that takes into account the angular aspect and significant impact this parameter has on the deployment within the network. In particular, this dissertation has made the following contributions.

- Model the general problem in a continuous space, striving to minimize the number of

required video sensors to cover the given 3D regions. We then address the problem by converting it into a discrete version where we incorporate 3D grids for our discrete model, which can achieve arbitrary approximation precision by adjusting the grid granularity. Consequently, by using the discrete model we can get more precise and realistic coverage space for each wireless video sensor.

- We design two strategies to tackle the additional challenges caused by obstacles, which are Divide and Conquer Detection Strategy and Accurate Detection Strategy. We demonstrate how both strategies can detect obstacles within the monitored space and can improve the performance of our solution by avoiding covering particles inside the shaded area caused by obstacles when we deploy the wireless video sensors into 3D indoor space. Consequently, we can get very precise and realistic coverage space for each wireless video sensor.
- Propose a mechanism to address k-coverage whereby we extend the discrete model by implementing a constraint for 2-angular coverage, so that each point within the area of interest is covered by at least two video sensors at opposing angles of a pre-defined value.
- Propose a scheme to ensure connectivity in the network among all sensor node continuously using a path protocol and a base station.
- Develop a Greedy-Heuristic algorithm that can achieve complete area coverage of the 3D regions by determining the candidate locations and directional angles to cover the maximum number of lattice points, thus deploying the locally optimal video sensor within the monitored area.
- Develop an enhanced Depth First Search algorithm that consists of an enhanced graph traversal method that searches the lattice of local candidate sites for optimal sensor node placement and angular direction. An area coverage function with a greedy heuristic

tic, a derived lower bound for search branch pruning and a simulated frustum culling method are also utilized to increase the efficiency of the algorithm and can produce optimal results if given enough time.

- Develop a customized Java Script 3D environment simulator to compare the proposed solutions. The results demonstrate the reduction in the number of required video sensors by the greedy heuristics achieves up to 50% over a baseline algorithm. Our enhanced DFS can achieve an additional reduction on the number of video sensors up to 37%.

1.3 Dissertation Organization

The remainder of the dissertation is organized as outlined below.

- Chapter 2 details an extensive overview of WSNs, the classification of WSNs deployment techniques, explores the significant impact of its evolution on WWSNs and discusses the current challenges associated with WWSN deployment.
- Chapter 3 discusses in more details the recent works related to the research conducted in this dissertation.
- Chapter 4 discusses the general deployment problem. The continuous and discrete space network models are defined in our problem formulation. Then, both the Greedy Heuristic and enhanced Depth First Search algorithms are proposed, where we present a detailed discussion on the pruning method to reduce the solution search space. The simulation environment is presented, where we introduce our Java Script 3D environment. A performance evaluation of each solution is provided.
- Chapter 5 presents an extension of both network models to include obstacle awareness and connectivity constraints. We explore both the Greedy Heuristic algorithm as well as the enhanced Depth First Search algorithm solution, whereby we discuss how

incorporating obstacles and connectivity into the 3D space impacted the solutions performance.

- Chapter 6 formally introduce the angular coverage problem. We discuss both the continuous space and discrete lattice based models revisions to include the 2-angular-coverage constraint in our approach to constructing the problem. The performance evaluation of both the revised Greedy Heuristic and enhanced Depth First Search algorithms are provided.
- Chapter 7 concludes the dissertation with additional discussions on the overall performance of the aforementioned algorithms, with some future directions for our work.

CHAPTER 2

BACKGROUND

A general overview is presented in this chapter for Wireless Sensor Networks (WSNs) and Wireless Video Sensor Networks (WVSNs). Section 2.1 will introduce the historical aspects of WSNs and discuss the current implications of the technology on the topic area. Building upon the general description of wireless sensor networks in practice, Section 2.2 will explore the evolution of the technology and present the different types of wireless sensor networks. In Section 2.3, an in-depth discussion of wireless multimedia networks is presented, addressing the broader concept of the networks in practice and some classifications of the category. WVSNs are reviewed with example applications in Section 2.4, where a comparison of traditional and non-traditional sensors is given. The history and current issues surrounding WVSNs are provided in Section 2.4.1 and 2.4.2, respectively. Finally, in Section 2.4.3 a description of common strategies for WVSNs deployment, connectivity and network issues is presented.

2.1 Wireless Sensor Networks

Although WSNs have a long and rich history, it only emerged as a prominent technology in recent years. Nowadays, this technology is easily integrated into many aspects of our daily lives. The proliferation of this technology can be seen in mobile, home utility (i.e, garage openers, television remote, etc.), commercial and medical products [29]. The technology is now more compact, affordable, portable and powerful (i.e. processing ability) than ever. This is a far cry of wireless sensors from days of old.

The origin of WSNs is heavily rooted in the development of technologies for government surveillance and military espionage in the 1950s. However, research as we are

accustomed to now using networks historically dates back to the late 1960s. During this time, a select number of universities within the United States begin to establish Advanced Research Projects Agency Network (ARPANET) programs to collaborate with government researchers and to develop internal projects [30]. In the early stages of this project researchers developed a packet switching network that was the first network to implement the transmission control protocol and internet protocol (TCP/IP) suite. The initial flagship institutions that participated in the ARPANET project consisted of the University of California -Los Angeles (UCLA), University of California - Santa Barbara (UCSB), Stanford Research Institute (SRI) and the University of Utah as illustrated in Figure 2.1, where the first successful attempt to send a message was established between UCLA and SRI.

WSNs are formally defined as a network of spatially distributed devices that are equipped with sensors to monitor large geographical regions or remote coverage areas (i.e. outdoor spaces) and environmental conditions [31]. These devices were established to work autonomously and are capable of self organization by logical linkages using sinks . Wireless Sensor Networks are categorized into a class of wireless ad hoc networks, where the wireless ad hoc network is a collection of wireless nodes [32]. The nodes have the ability to communicate directly over a shared wireless channel. There is no additional infrastructure that is required for ad hoc networks. So, every node within the network is equipped with a wireless transceiver that can process data packets and guide them to their destinations. As a result of these features, the use of wireless sensor networks in research grew exponentially, and extensive research has been conducted in the area of Wireless Sensor Networks (WSNs) to provide many beneficial techniques. WSNs are now applicable in numerous areas including: industry, medical, networking, environmental and transportation fields. There are many advantages associated with the use of WSNs such as the economical cost to setup a network and scalability within the network. As a result, WSNs have expanded into several other categories of sensor node networks.

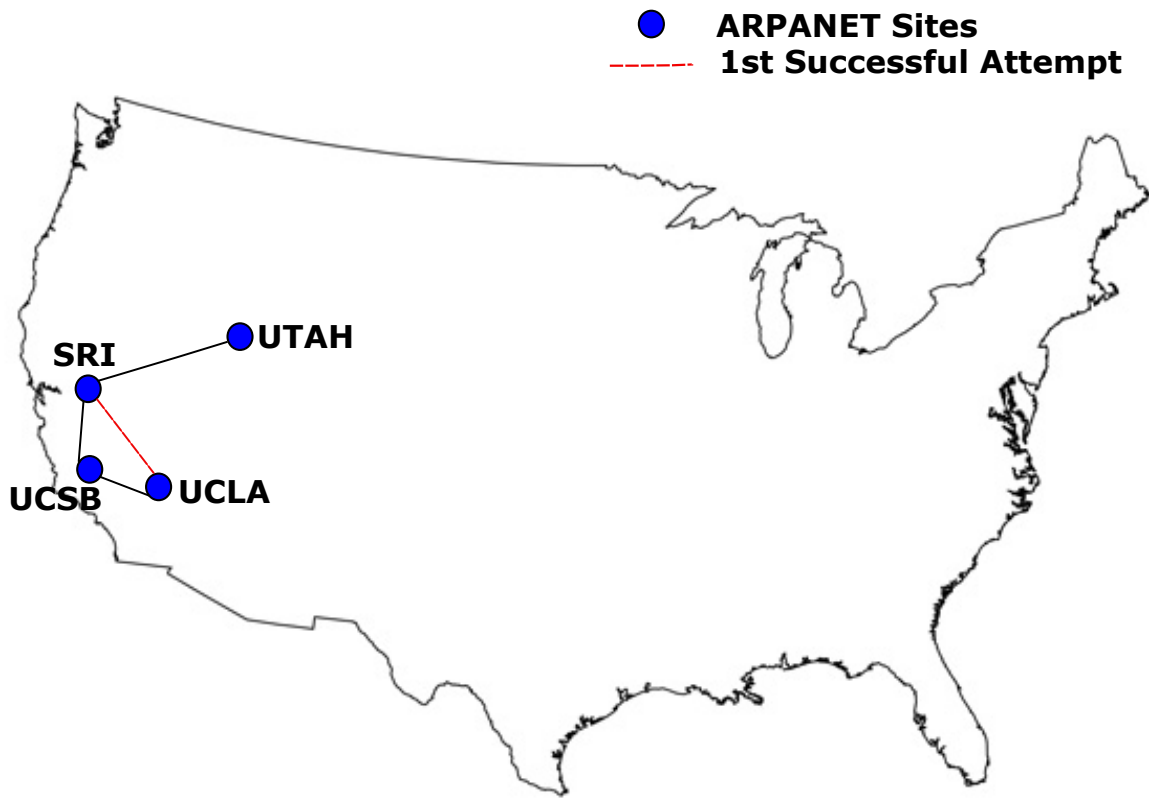


Figure 2.1. The Beginning of ARPANET [1]

2.2 Evolution of Wireless Sensor Networks

In the late 1980s, the high rate of demand for Wireless Sensors Networks was at an unprecedented level. Initially, the use of WSNs was primarily for private industrial applications, government entities and military science. The true potential and capabilities of Wireless Sensor Networks began to be explored more in the 1990s when joint efforts among academia and industry initiated efforts to solve challenges within Wireless Sensor Networks. The challenges associated with growing the technology were centered on the bulkiness, high cost and patented protocols of the sensors. This all changed due in part to the advancement of fields such as material science, networking and embedded processors. Initiatives among

universities and commercial industries such as University of California-Los Angeles (UCLA), University of California-Berkeley, National Aeronautics and Space Administration (NASA) and ZigBEE assisted in the identification of core functional problems within the network that included: reducing the cost per sensor, extending the energy of sensors and enabling deployment in commercial applications.

Wireless Sensor Networks have played an integral role in countless technological areas. The sensors of yesterday have been transformed into state of the art devices that provide Wireless Sensor Networks a new generation of features to offer to consumers. In some cases, sensor costs have declined by as much as 100X times over the past decade as seen in Figure 2.2, where a chart illustrates the decrease in cost of sensors over the decades (1950s-2000s) in various application fields.

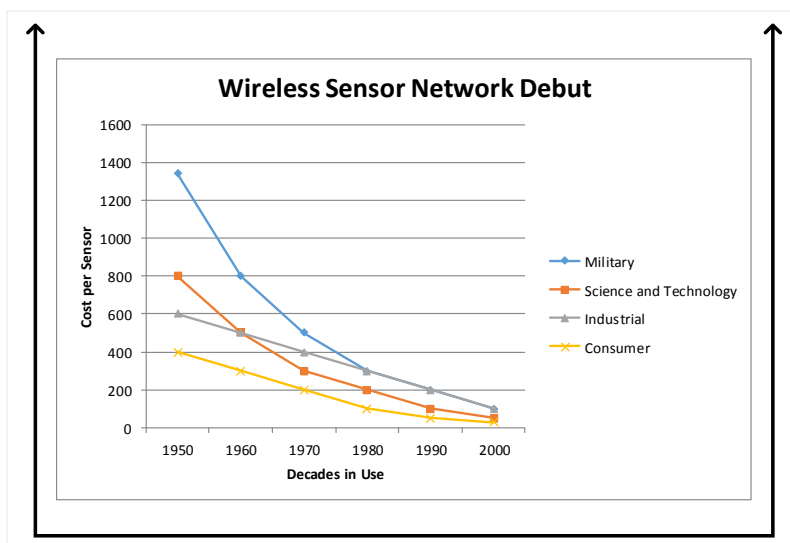


Figure 2.2. Evolution of Sensor Cost [2]

Additionally, Figure 2.3 provides a bar graph that showcases the gradual market gains in consumer applications as compared to military, science and industry over the past decades (1950s-2000s) as reported by Silicon Labs [2]. Additionally, the chart outlines the

transitional change from the core military applications in the past to the increase in industry and consumer applications more recently.

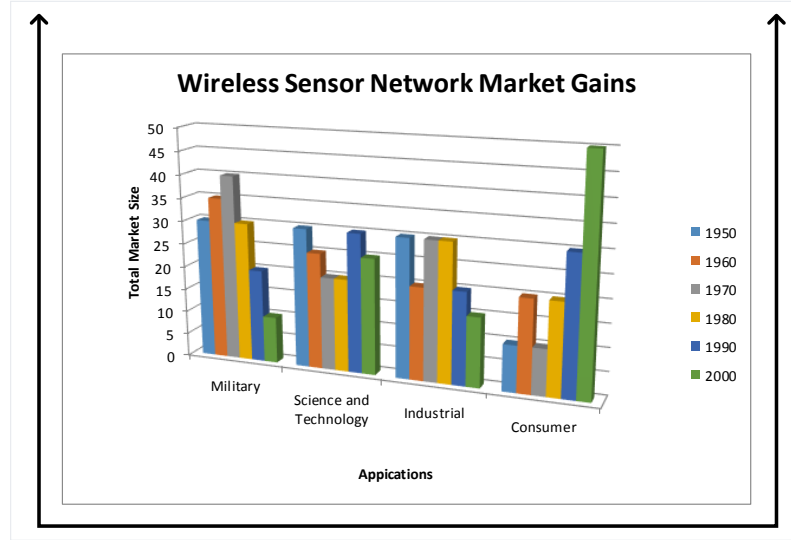


Figure 2.3. Applications Market Gains [2]

Consequently, with the popularity of WSNs many subcategories of Wireless Sensor Networks have emerged. Some of the subcategories identified as a type of Wireless Sensor Networks include: mobile, underground, underwater, terrestrial and multimedia WSNs [17] [33], as illustrated in Figure 2.4. The following provides a overview description of each WSNs type mentioned above.

- *Wireless Mobile Sensor Networks (WMSNs)* are compact portable sensors that can move with their carriers to patrol and monitor any environment. These sensors are suitable for scenarios where traditional deployment techniques such as manual or air drops are inapplicable. WMNs play a critical role in security applications and are often used in emergency first responders scenarios (e.g., policemen, fire fighters, medics) [34].
- *Wireless Underground Sensor Networks (WUSNs)* provide an efficient wireless commu-

nication in the underground medium. The propagation medium consists of sand, soil and rock. This type of sensor network is useful for applications that include soil condition monitoring, earthquake prediction and border patrol. Magneto-inductive (MI) waveguide techniques are commonly proposed to cope with the very harsh propagation conditions in WUSNs [35].

- *Wireless Underwater Sensor Networks (UWSNs)* are essential to numerous underwater applications, including underwater tracking and tactical surveillance for endangered sea life. Under water acoustics sensors are used primary to estimate the long dynamic propagation delays in data transmissions. UWSNs are beneficial in a wide range of applications including: wide range of aquatic diving, coastal-line surveillance, underwater exploration and tsunami prevention [36].
- *Wireless Terrestrial Sensor Networks (WTSNs)* are multifaceted sensors that allow for use above as well as underground. WTSNs use solar cells and low duty cycles to conserve energy. The sensors are deployed usually in an ad-hoc manner. The sensors play an active role in applications involving surface exploration, environmental and industrial monitoring [37].

2.3 Multimedia Wireless Sensor Networks

Multimedia Wireless Sensor Networks (MWSNs) are a category of WSNs that allow for the monitoring of events in a multimedia format, such as images, videos, and/or audios [38]. The MWSN consists of low-cost sensor nodes equipped with microphones and cameras. The nodes are often interconnected with each other over a wireless connection, which allows for compression, retrieval and correlation of multimedia data. In contrast, wireless sensor networks are deployed for physical phenomenons such as temperature, pressure, humidity or the object's location which is essentially numeric data [39]. These general applications provide low bandwidth demands and are normally delay tolerant. However, Wireless Mul-

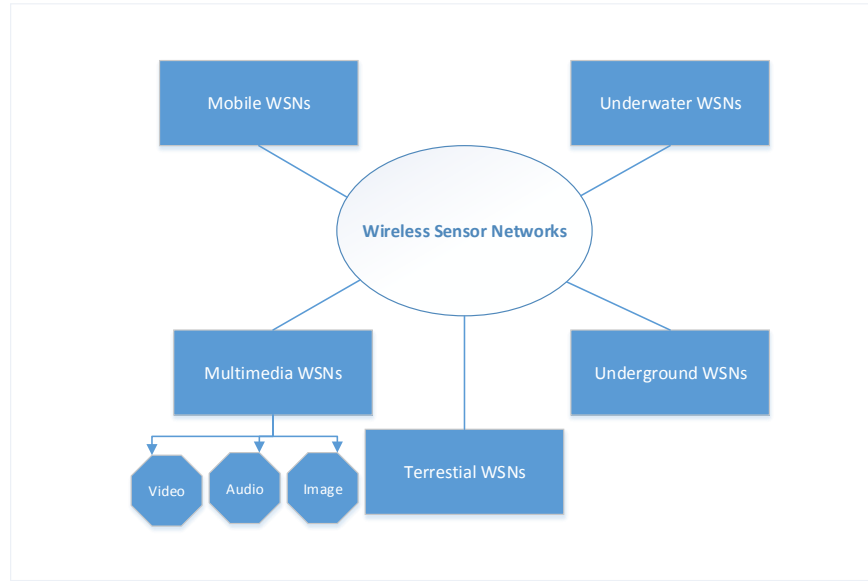


Figure 2.4. Types of Wireless Sensor Networks

multimedia Sensor Networks pose a greater challenge in terms of deployment but can provide enhancements to the existing sensor network applications which includes: tracking, home automation and surveillance monitoring. Traditionally, WSNs allow sensor nodes to retrieve information from the environment with a predefined sensing range, i.e., a circular disc with a range defined by the type of specified sensor. Figure 2.5 highlights an example of the traditional sensing range models of WSNs. In these models, the sensing range is omnidirectional with 360° sensing capabilities. On the other hand, multimedia sensors primarily have larger sensing ranges and are also sensitive to the directional angle of the sensor's position. In particular, cameras can capture images of objects or parts of regions that are not necessarily close to the camera. These unique features identified within MWSNs are beneficial in 3D indoor environments because of the practical and real world applications that can be improved upon including: campus, industrial, medical and security monitoring. Furthermore,

with the introduction of these techniques the efficiency of MWSNs can become independent of the existing network infrastructure (i.e., building's indoor network infrastructure) allowing direct connectivity among the deployed indoor video sensors.

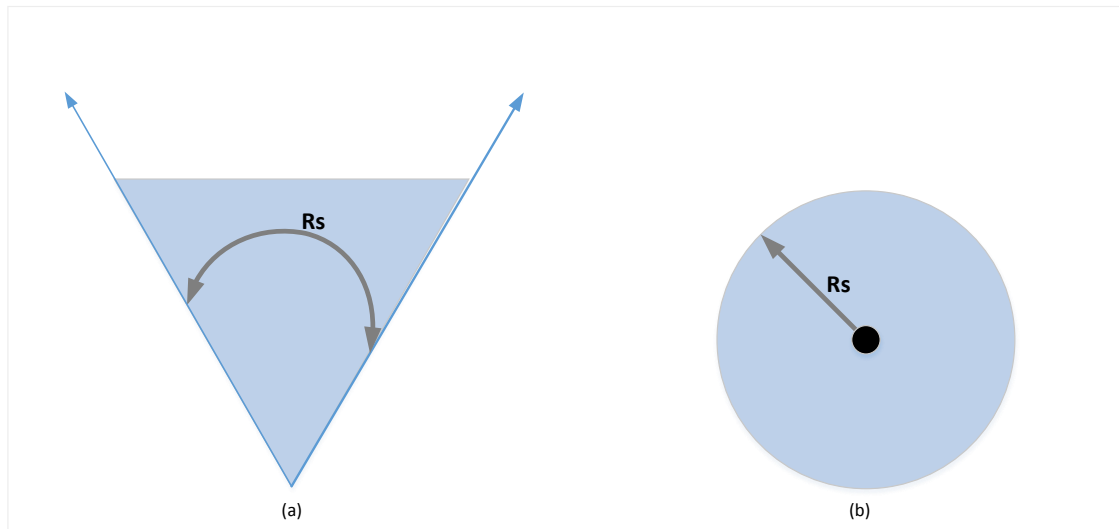


Figure 2.5. Traditional WSN sensing range models. (a) 2D directional sector coverage model; (b) 2D omnidirectional sensing range model

2.4 Wireless Video Sensor Networks

Over the course of a decade or so, WWSNs have emerged as a leading technology applicable among many fields in government, academia and industry, due to the economical cost of complementary metal-oxide-semiconductor (CMOS) and the versatility of signal processors [40]. A wireless video sensor network is a type of wireless multimedia sensor network that has cameras mounted on distributed sensor nodes that allow for the recording of digital images [41]. This technology has unlimited potential for numerous application fields. However, wireless video sensor networks have introduced new research challenges to the field of WSNs. Wireless Video Sensor Networks are comprised of a set of nodes where each node typically consists of four major components: a micro-controller, sensor, power supply and a

wireless communication device. In this instance the sensor utilizes a camera as an interface attached to the node to view the physical world [42].

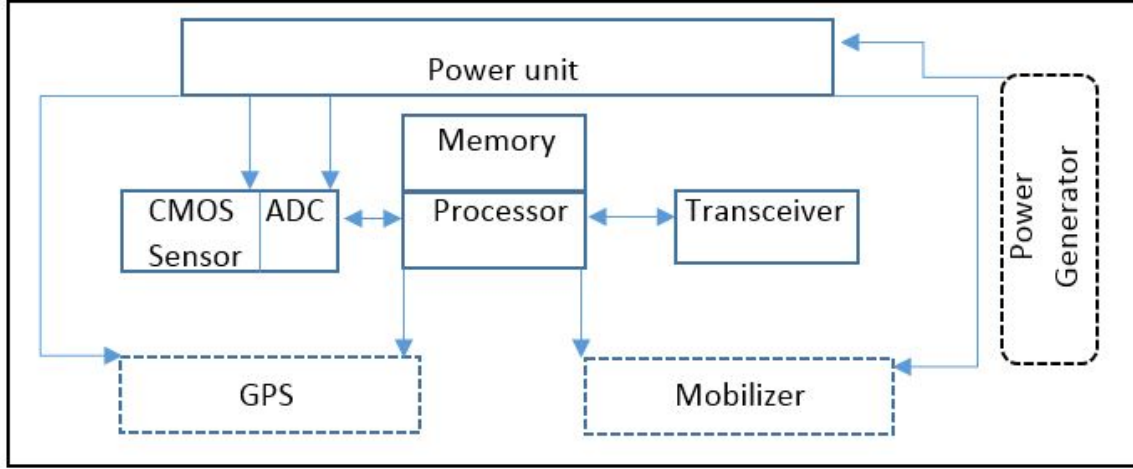


Figure 2.6. Components of a Wireless Video Sensor Node

A sensor node within the wireless video sensor network is capable of carrying out such tasks as: processing data, communicating with other nodes within the network and collecting sensory data from the environment [43]. These types of sensors can range from a simple camera with location and position properties to a high resolution image sensor that can perform facial recognition. The basic components of a video sensor network are shown in Figure 2.6. Every component of the sensory board serves a specific purpose. The micro-controller is essentially the “brains” of the unit in the node. The primary purpose of the micro-controller is to control the sensor and gather information. It is also responsible for processing the collected information. The micro-controller is normally equipped with external or internal memory storage to collect and process information. Also, the micro-controller controls the wireless communication device and decides when and where to send information. It can also receive data from other nodes and decide what to do based on the content of the received data. The power source for the sensor is often an external battery that is connected to the board. The antenna serves as a communication extension to provide

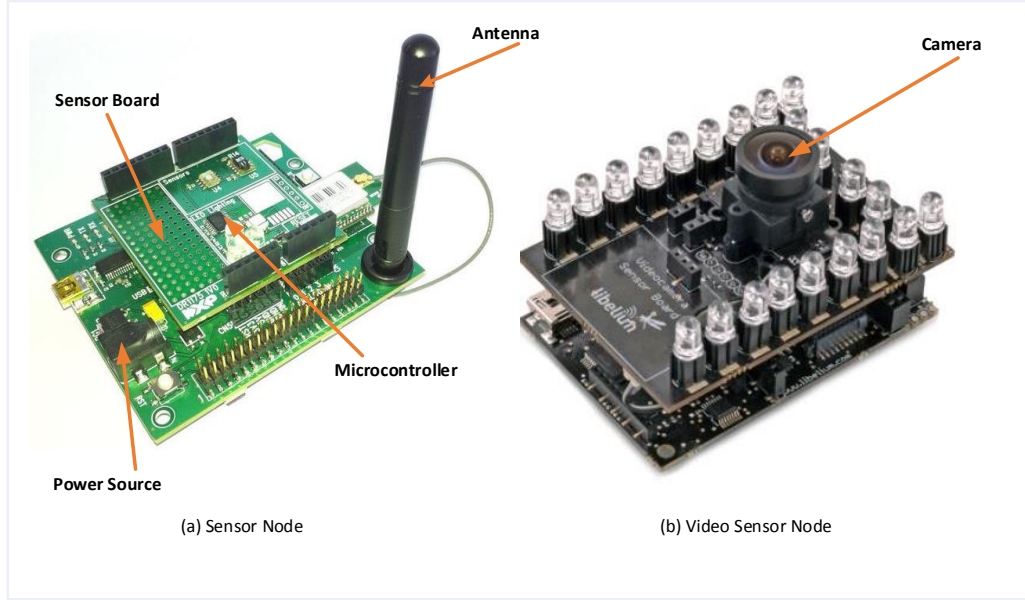


Figure 2.7. Sensor Node Comparison [3]

access between individual sensor nodes. Figure 2.7 illustrates the components of a traditional sensor node versus that of a video sensor node that has a camera attached as seen in [44]. The sensors in general have very limited resources in terms of energy, coverage and connectivity.

2.4.1 History of WVSNs

Wireless Video Sensor Networks(WVSNs) fit into a category of network platforms similar to the technology acknowledged in Wireless Sensor Networks. The early concept of WVSNs originated in the 1960s. The primary usage was to serve as a monitoring application for Closed Circuit Televisions (CCTV). These types of networks were controlled by the government and other proprietary organizations. Currently, WVSNs have transitioned into a more practical consumer minded approach where smaller network devices are used to capture and monitor home or office spaces using standard IP based platforms. The surveillance driven

market is now at the forefront of corporate innovation and social media research interests [45]. The majority of the market caters to tech surveillance companies and the millennial generation that crave documenting and over-sharing detailed events of their lives. The large amount of video data poses a QoS (Quality of Service) issue during transmission due to traffic delays within the network.

2.4.2 Issues in WVSNs

Wireless Video Sensor Networks are capable of delivering many positive features, though there are issues and trade-offs that are associated with the development of a reliable WVSN. In this section we discuss some of the factors surrounding the challenges that exist in the deployment of wireless video sensor networks. Some of the factors were mentioned earlier in previous sections of this chapter, however, what proceeds will touch specifically upon sensor deployment issues and other inherited limitations of WVSNs. A deployment scheme addresses the placement or arrangement of sensors within a network that can influence several performance metrics, including coverage, connectivity, obstacles and network traffic [46] [47].

One of the central issues associated within WSNs and WVSNs is the coverage problem. In WVSNs, coverage can have numerous meanings and can be explored using different methods. In general, Deif and Gadallah [19] posed it in a question to determine "How well do the deployed sensors observe the physical space?". When classifying coverage in WSNs it is divided into three types as seen in Figure 2.8: area (blanket) coverage, point coverage (covering a set of finite points in a region of interest) and barrier coverage (detecting movement across a barrier of sensors) [48] [19].

The coverage problem focuses on two components: (1) maximizing the number of points in the monitored region covered by the sensor's sensing range, and (2) minimizing the number of sensors required to cover the entire monitored area. This is a major factor that contributes to the performance optimization of the network and is a highly regarded research area in WSNs. Likewise the deployment problem is a major issue in wireless video sensor

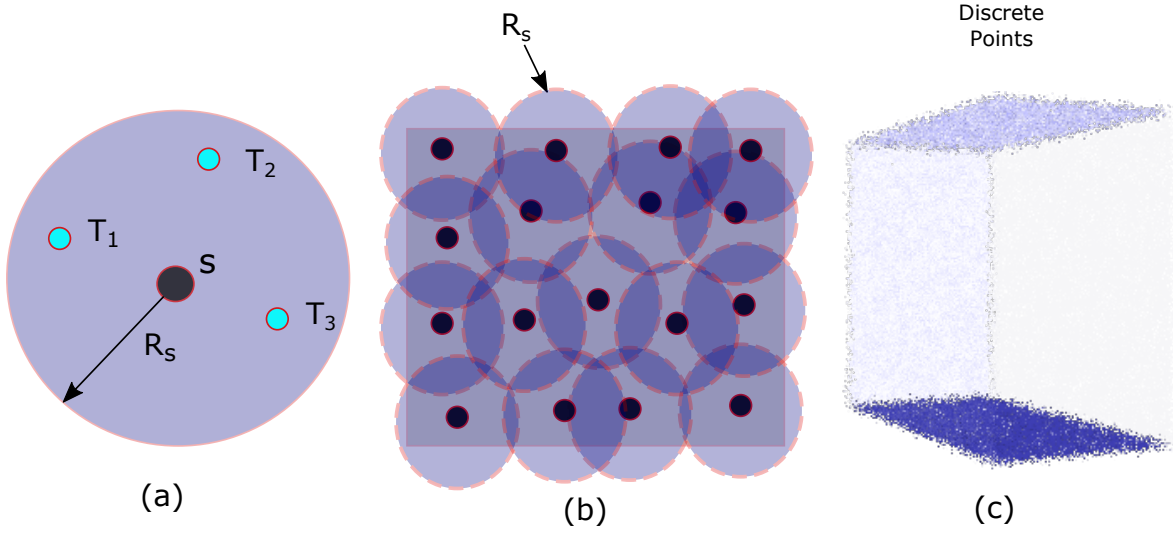


Figure 2.8. Coverage Classifications (a) Target Coverage (b) Area Coverage (c) Barrier Coverage using pointillism of discrete points [4]

networks. In previous studies, a general model of the problem is given where several sets of points within a monitored region are defined as a cover-able set. A sensor is then deployed (placed) in the space with parameters such as a sensing range, position (sensor location) and orientation. A point is considered to be covered if it is within the sensing range of any of the video sensor. Additionally, a deployment scheme is considered efficient if it reduces the number of video sensors deployed in the monitored space required to completely observe (cover) the region.

However, other factors that weigh heavily on the performance of the WVSN are often overlooked such as the quality of the deployment scheme. When analyzing the quality of coverage within wireless video sensors networks a different metric is needed to measure the quality of the deployment in terms of coverage. Often times the k -coverage (i.e., every point is covered by at least k sensors) metric is applied.

Another important factor in WVSNs is the assurance of connectivity within the net-

work. The sensors' ability to communicate with each other is of high priority. Many networks try to support 1-connectivity or 1-hop approach property where the data can be routed to one node and reach its destination (i.e., the sink) within the network. This is based on the assumption that the WWSN connectivity is primarily determined by the sensors deployment locations and the communication ranges. The objective is to develop a communication model where the sensors are relatively clustered promoting the rationale of each sensor being within the communication range of its neighbor. Many related works reviewed formulated the problem of wireless sensor connectivity modeled as a directed graph. In the terms of directed graphs, there are three types of connectivity as stated in [49]. Figure 2.9 illustrates a representation of a strongly and weakly connected graph as defined below.

- Weakly connected - if replacing all of its directed edges with undirected edges produces a connected (undirected) graph.
- Regularly connected - contains a directed path from distinct vertices u to v OR a directed path from distinct vertices v to u for every pair of vertices u, v .
- Strongly connected - contains a directed path from distinct vertices u to v AND a directed path from distinct vertices v to u for every pair of vertices u, v .

One of the fundamental and most commonly addressed factors in WWSNs is that of traffic delay. Due to the immense amount of video data retrieved and transmitted within the WWSN, traffic flow within the network is a major issue. The avoidance of traffic delay in the network, where a scheme to relay data in an optimal path increases the performance is crucial. QoS-aware networks where end to end delivery protocols are introduced into the system to provide management and scheduling algorithms are desired [50].

Finally, the issue of addressing obstacles within the monitored area in WWSNs is a critical factor. In an indoor environment, there are often obstacles such as desks or furniture, which introduce additional challenges and further render the deployment solutions for

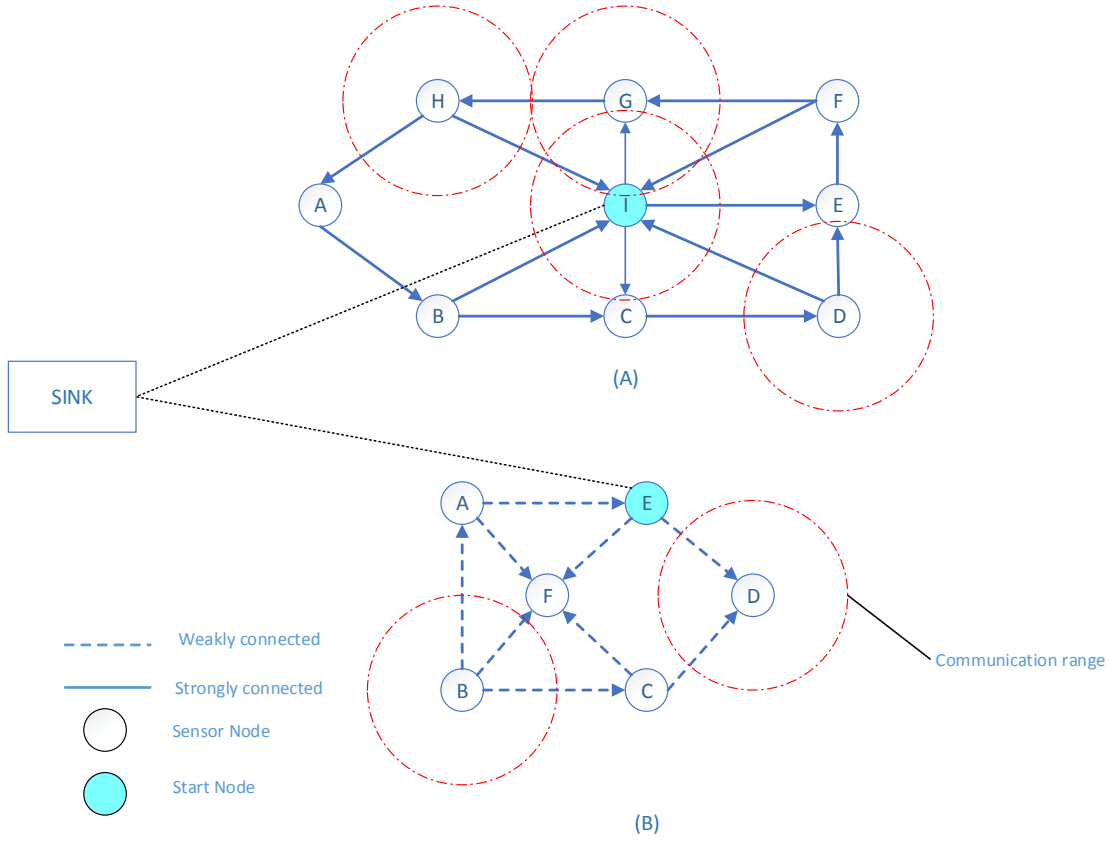


Figure 2.9. Connected Directed Digraphs (a) Strongly connected graph (b) Weakly connected graph

traditional sensors and 2D sensing fields obsolete. For example, considering obstacles within the monitored region helps to reduce the allocation of sensor resources that may be deployed for areas (monitored points) that are blocked by obstacles. The ability to avoid these points that are of no consequence to the improvement of the network coverage will improve the overall performance.

2.4.3 Strategies

In this subsection, discussion is provided on some of the common strategies used to solve deployment, connectivity and networked issues in WVSNS.

Wireless deployment strategies are either considered to be planned or random de-

ployment. The random deployment approach is the most commonly used tactic to deploy sensors because it is easier and less expensive for large wireless sensor networks [51]. Most of the works in literature are focused on randomly deployed video-based wireless sensor networks. However, in a 3D indoor space, it is more reasonable to adopt planned deployment (i.e., deterministic deployment), which is what we will focus on in this dissertation. The WVSNs that we are currently undertaking involves sensor nodes that have the ability to sense its environment and communicate with each other. In this section, we consider commonly used deployment strategies. The three categories that are often used to deploy sensors are computational-geometry, force, and grid based [52] as shown in Figure 2.10. In force based strategies deployment is dependent upon the mobility of the sensors, specifically noting where energy forces can repel or attract the sensors within the network. This approach assumes a virtual force theory proposed by Khatib [53], where nodes have a virtual force property which can attract sensor nodes to each other and transmit data. Often times the sensor node is considered a mass and using a formula, the virtual force of each node can be calculated [54]. This approach is attempted when full coverage is needed within the network. The computational-geometry based approach is employed often times when optimization is desired in the deployment scheme. In the computational geometry based approach, geometry is used to evaluate the problem and define an algorithm. Using this approach, the algorithm is designed to efficiently solve the problem using data structures and basic geometric objects such as: line segments, points and polygons. Methods based on computational geometry include: “Relative Neighbor Graph, Gabriel Graph, Voronoi Diagram and Delaunay Triangulation” [6]. Using these techniques every node is assigned enough power so that each node can reach its neighbor in one hop.

Grid based approaches are implemented in sensor deployment to assist in the sensor positional tracking and as a way to determine the coverage efficiency using point allocation. To apply this approach, the grid can represent the layout of the monitored area and then the problem will be modeled and formulated as the area of a grid. An algorithm is then

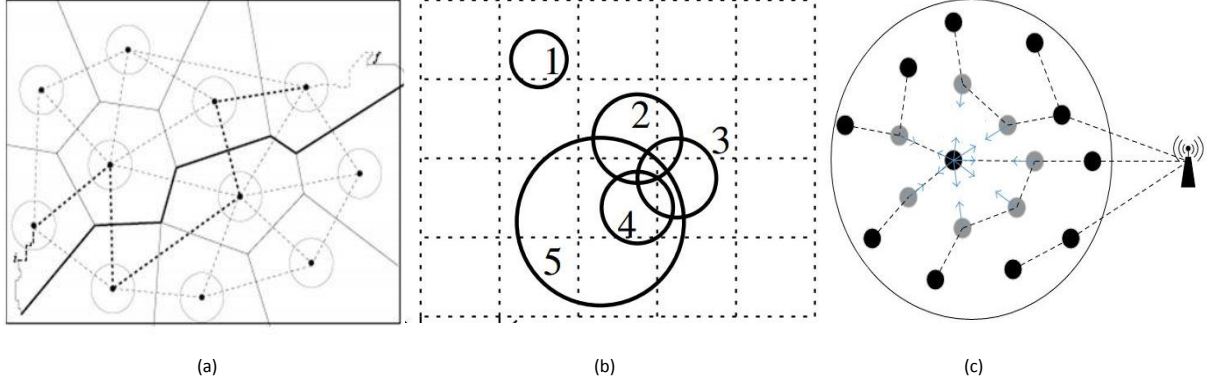


Figure 2.10. Common Deployment Strategies (a) Computational Geometry-based; as shown using Delaunay Triangulation (b) Grid-based (c) Force-based as shown using Virtual Force theory [5; 6]

based on the modeled grid. Essentially, the coverage area of a sensor is represented by a set of intersection points of the grid.

After the deployment of wireless video sensors, several studies [55] [51] suggested the implementation of coverage metrics to measure the quality of the calculated coverage in the network. Within the survey [51], the authors identified three reasonable metrics to measure the deployment with consideration to coverage. The first metric is referred to as the Network Coverage Area (NCA). Using NCA, the node's area is defined and the relevant sensing area of a node is denoted as a sector of a circle. This sector represents the intersection of the sensor node field of view and node area. Using a function termed Deployment Coverage quality (DCQ), shown in equation 6.2, the ratio of total relevant sensing areas (NCA) and the collective sum of all relevant sensing areas (NRSA) are evaluated.

$$DCQ = \frac{NRSA}{NCA} \quad (2.1)$$

Another widely proposed coverage metric is k-coverage. The k-coverage problem generally states that any point within the monitored area must be located in the sensing range of at least k-sensors. As an example, if the deployment area is defined as 4-coverage, every point within the area is covered by at least four sensors. This means that up to three of the four nodes sensing the same monitored region can fail and the area will still be covered (monitored) by one of the four nodes. Full k-coverage denotes that every point is covered by all k-sensors and where k is usually defined by users as greater than or equal to one. It is often suggested that guaranteeing 100% coverage in WVSNs is very difficult in a random deployment scenario. So the authors in [56] proposed a directional k-coverage metric (DKC). DKC demonstrates the coverage quality measured in terms of a probability guarantee. It utilizes a summation function to calculate the probability of coverage for a target by a camera considering the video sensor's field of view, deployed nodes and sensors covering the same region.

The following paragraph below explores some common connectivity strategies utilized in Wireless Video Sensor Networks. Coverage in a WSN is intertwined with the connectivity in the network as advised in [19]. Consequently, the same can be assumed for the coverage-connectivity relationship in WVSNs. Connectivity among sensor nodes can be established using direct or indirect pathways. As seen in traditional coverage approaches, the connectivity topologies for WVSNs are often modeled using graphs. Given a graph \mathbf{G} , where $G = (V, E)$. We can denote V as a set of sensor vertices and E as set of communication links connecting neighboring sensors. Given this formal definition, the strategies for communication among sensors can be divided into two types: node connectivity and edge connectivity. The node connectivity method outlines the connection directly using node pairs. If a path exists between the node pairs in a single or multi-hop scenario, the graph is connected; otherwise the graph is not connected [57]. Similarly, when there are at least k edge disjoint

paths between every pair of nodes, there exist a path to communicate among the nodes. Once the connectivity of a network is established, factors such as path losses, capacity and link-quality need to be considered. These metrics allow for the analysis of the signal strength between sensor nodes measured as Signal-to-Noise Ratio (SNR) or Received Signal Strength Indicator (RSSI) [58]. In a directed graph, an edge connects two vertices. If the vertex is removed from the graph, all of the edges associated with that vertex must also be removed. These edges can not arbitrarily be connected to other vertices.

The author in [59], investigates the connectivity problem for wireless multihop networks considering homogeneous random node distribution. In the study, the author proposes a geometric random graph model for the network to represent the wireless communication links (i.e., edges). The work focuses on identifying the graphs minimum node degree and its k -connectivity. The derived analytical expression can determine the the required transmission range rO . For the given node density parameter, an adequately k -connected network is ensured using the edge connectivity. The problem evaluation utilized simulation in various scenarios, with and without border effects. Hence, the node and edge issues become increasingly important for the overall performance of a WWSN monitoring infrastructure.

CHAPTER 3

RELATED WORK

This chapter provides an overview of research related to this dissertation. More specifically, Sections 3.1.1, 3.1.2 and 3.1.3 provide a review of related studies for WSNs considering deployment, coverage and connectivity, respectively. Also included is an overview of past and existing research studies on WVSNs for deployment, coverage, connectivity and obstacle detection in spatial environments to provide a reference point outlining the gradual progression in techniques used to resolve the aforementioned issues. Several studies have addressed the deployment problem in WVSNs by proposing solutions regarding the coverage and connectivity issues in 3D environments. Based on these studies, we address the related works classified into two categories for 2D and 3D WVSNs. Section 3.2.1 discuss current studies that address the deployment problem for WVSNs. Section 3.2.2 presents existing research studies that address the coverage problem in WVSNs. In Section 3.2.3 a summary of related studies that address the connectivity problem for WVSNs is detailed. Finally, a review of work considering obstacles awareness in WVSNs is provided in 3.2.4.

3.1 Wireless Sensor Networks

The term sensing is formally defined as the ability to perceive or gather information about a physical phenomena [60]. The device performing the sensing is referred to as a sensor. In WSNs, sensors have the unique ability to not only monitor numerous environment phenomena but also provide information about the physical properties of that environment including: pressure, temperature, humidity and light; without significant human intervention. However, there are several metrics that are crucial to improving the overall performance

of the network. A review of related work on deployment, coverage and connectivity in WSNs is presented in the following subsections.

3.1.1 Deployment

The deployment problem has been widely explored in WSNs for 2D environments [61] [62] [63] [64] [65] [66]. Here, deployment is concerned with the placement or positioning of a sensor network for effective action in a real-world environment. Often times, deployment is a labor intensive task as environmental hazards (i.e., bugs, wind, rain) can degrade performance in outdoor environments and indoor restrictions (i.e., wall, ceiling space constraints) can factor into placement of sensors [67]. Hence, effective deployment strategies are useful in maintaining the functionality of the sensor and obtaining high quality performance within the WSN.

Traditional approaches for WSNs are categorized as 1) *random* and 2) *deterministic* schemes. A key determining factor in selecting the type of approach for deployment is the deployment area. In a scenario where random deployment is used, often times factors such as scalability for large-scale open regional monitoring is emphasized. However, in small-scale deployments a deterministic strategy is usually chosen for the placement of wireless sensor nodes [68].

In [69], the authors proposed an approach where a 2D model is implemented for the planned deployment of sensors. Utilizing a linear hierarchical network the authors propose a scheme to minimize the number of nodes needed to construct an efficient network by developing optimization algorithms where the greedy solution considers distance to the sink from the nodes. Presented in [70] is a control-based sensor deployment algorithm for detection and surveillance in WSNs. Authors address the deployment problem by determining the locations in order to satisfy the detection requirements for a squared error. The problem is modeled as a dynamic system and formulated as an optimal linear quadratic regulator control problem. Two solutions are given, namely an optimal control based and Max Defi-

ciency algorithms, both of which only offer nominal gains, the former being computationally demanding due to the implementation of a sweeping method.

As previously stated, the number of sensor nodes deployed within an environment has a direct impact on the cost and the performance of the network where a robust and fault tolerant environment is desired. As a result, authors in [71] evaluated several sensor deployment techniques to determine the efficiency (i.e., determine the pattern that requires the least number of sensors with the smallest overlapping occurrence to cover the monitored space) of each based on a coverage area ratio. The work focused on *geometric pattern deployment* within a grid, specifically evaluating triangular, square and hexagonal lattices. Objectively, the focus of many studies which employ geometric deployment propose to maximize the coverage percentage and ratio of area covered by at least one sensor relative to the total area of the region of interest (ROI). The study concluded the triangular lattice pattern performed the best among the other deployment options.

As is common practice in many previous studies, WSNs are traditionally formulated to use 2D models for scalar sensors, all the studies highlighted above follow that mold. This is typical of previous studies because of the inherent difficulty in studying the deployment problem for 3D coverage and connectivity. The problem is considered to be NP-hard even without integrating obstacles into the monitoring environment. In a rare attempt to address the 3D deployment problem for WSNs, the authors in [72] propose a deployment planning tool to minimize the cost to achieve full coverage and node connectivity in a 3D target area with obstacles. A 3D model is formulated for the deployment space with point and cost variables. A heuristic algorithm for computing a near optimal solution is proposed. A smart space simulator *UbiREAL*, is presented to implement an interface module for configuring the sensor deployment within the space. However, there is no quantitative data or benchmark testing presented to verify the performance of the solution. A figure of the simulator test is simply given.

A relatively new technique proposed for sensor deployment is clever *swarm intelligence*

where algorithms are implemented using the collective behavior patterns of self-organized systems, natural or artificial such as ant colonies, animal herds schooling fish or even bacteria growth [73]. Authors in [74] propose an Artificial Bee Colony (ABC) algorithm for the dynamic deployment problem in WSNs. A probabilistic detection model integrates a scenario of mobile and stationary sensors. The ABC algorithm is a swarm intelligent technique based on the foraging behavior of honey bees. The algorithm utilizes food sources which correspond to placement positions and the nectar amount of the food source represents the quality fitness of a candidate solution. The evaluation of the ABC algorithm is compared with a baseline particle swarm optimization algorithm (PSO), also another well-known swarm based optimization technique.

Similarly, authors in [75] propose an Ant Colony Optimization algorithm to solve the deployment problem for a WSNs. Using a colony of artificial ants behavioral patterns a mathematical formula is proposed. These cooperative agents then search a simulated environment to find optimal paths. The environment is represented as a simple graph. Highlighted in the ACO algorithm is a max-min ant system (MMAS) where the parameter uses a fixed upper and lower bound of pheromone trails to prevent the redundancy of traveled ant pathway solution sets.

3.1.2 Coverage

Coverage is a fundamental issue in improving the performance of WSNs. The coverage metric is an indicator of how well each sensor deployed within the network can monitor the coverage area. Several studies have surveyed the deployment problem in WSNs proposing solutions regarding the coverage metric in 2D environments [76] [77] [78]. Typically, sensor node models for coverage are categorized into three different types including: index, binary and probabilistic models [79]. An extensive survey of the coverage problem for WSNs is presented in [80]. The authors explored various aspects and challenges that exist for WSNs coverage, including: classifications of coverage, types of network deployment, node models,

target characteristics, applications attributes and types of monitoring space. Accordingly, the related works discussed below are classified by the 1) type of monitored area that will be covered (i.e., barrier, point, regional) and 2) the type of deployment employed (i.e., random or planned/deterministic) for 2D WSN. Presented below is a review of related work for the coverage problem as it pertains to barrier, point and regional coverage.

The concept of barrier coverage was first introduced in the context of robotics sensors. The idea behind barrier coverage is essentially to detect a breach or intrusion attempt in a region for the entire coverage area. In [81], Liu et. al study the strong barrier coverage problem using a randomly-deployed sensor network. Contrary to previous works that highlight the critical conditions of weak barrier coverage, this study focuses on ensuring that intruders can not breach or cross the barrier strip undetected independent of the method chosen for path crossing. The authors propose theoretical foundations and a practical divide and conquer algorithm that divides the barrier strip into small segments. The algorithm can then compute the vertical barriers into vertical strips. In the next phase of the algorithm the horizontal barriers are computed in each segment connected by the vertical barriers into the neighboring vertical strips. A width-to-length ratio (i.e., the logarithm of the length) beyond which strong barriers start to emerge in the strip is calculated. An efficient distributed algorithm is also proposed to construct disjoint barriers in the sensor network. The strategy for weak barrier coverage is commonly used because of it guaranteeing the detection of intrusion for crossing the barrier strip only along an orthogonal path, an easier implementation method for detection. However, in the case of strong barrier coverage, a guarantee for intrusion detection is given no matter the path followed by the target (i.e., intruder), which requires a more complex approach.

Point coverage is a method used to model the coverage problem using a discrete approach. Point coverage for WSNs has been extensively explored [82] [83] [84]. Point coverage states, given a point P in a monitored space, it is covered if it is within the sensing range of a sensor. In Yang et. al [85], authors argue the case for guaranteed area coverage (i.e.,

regional coverage) by approximating point coverage for a WSN. The model in the study assumes the network is densely populated, thus the point coverage can simulate full area coverage. The objective of the study is to minimize the energy cost in the network while maintaining k -coverage. The proposed solutions include the construction of a dominate set based on the traditional graph theory. The authors propose several solutions, one global and two non-global algorithms. The global solution implements a cluster-based approach to select backbone nodes to form the dominate set. The non-global solutions use a pruning algorithm based on 2-hop neighborhood information. To evaluate the performance the authors employ both theoretical analysis and simulations. In [86], authors explore the coverage problem modeled as grid points for sensor detection. An effective full area coverage (i.e., regional coverage) and communication protocol is given. The authors use homogeneous scalar sensors to model the problem in 2D environments which can result in the solutions not being applicable to real world coverage issues. The coverage problem is indeed a fundamental issue in WSNs, so a comparison study to understand its metric and the independent impact of both regional and point coverage is addressed in [87]. Additionally, a current patent [88] is pending for a system that generates virtual interest points in an image, accordingly to detect points of interest in an image detected by an interest point detector using point coverage. In [89], the authors proposed a scheduling method which addressed the k -coverage problem to extend the network lifetime of WSNs. In their approach a simple 2D model with an omni-directional sensing field is employed.

3.1.3 Connectivity

Connectivity in WSNs provide a gateway for wireless communication back into the wired world [21]. The concept of connectivity in WSNs can be simply defined using a communication topology modeled as a graph for a WSN where the graph is denoted by $G = (V, E)$ and V is defined as a set of sensor vertices whereby E is denoted as a set of wireless communication links. A sensor pair is then considered to be neighbors of each others if and

only if their distance is at most the communication range c_r , considering the disk communication model. Theoretically, the concept of WSNs is easily stated. However, the true implementation of full connectivity is a challenging and lengthy process though a necessary one. For example, if a WSN within a monitored space relies only on existing WIFI availability, there are circumstances that may occur where the system fails or become unavailable (i.e., power outage). So, the handling of large amounts of scalar data (i.e., temperature, humidity and pressure readings) that are transferred frequently by each sensor node is of great importance. One of the most significant questions regarding the connectivity of WSN applications is the paramount concern to address the need for continuous connectivity that provide communication links among all the sensor nodes. Additionally, many WSN applications are continuously dependent on the reliability of their connectivity schemes. Natural environments in sparse regions like tropical jungles where researchers have a WSN deployed to monitor the habits of primate monkeys, desert land deployment to monitor shifting sand storms and oceanic seabed monitoring of coral reefs growth environments are sample cases of scenarios where frequent disconnections can cause the loss of critical data collections or even miss notifications of an impending disaster warning. Based on these concerns, related work is presented which discuss the impact of connectivity in communication dependent environments.

As an example, environmental monitoring often requires continuous operation for months [90]. Infrastructure monitoring (i.e., highways and bridges) requires an operational lifetime of months or several years. Authors in [91], investigate how the placement of relay nodes in a forestry environment of WSNs will impact the connectivity of the network. The authors identify a concept using algebraic graph connectivity to indicate the minimum number of relay nodes and links whose removal would result in a weakly connected graph. The goal of the study is to minimize the weighed link cost for each of the communication pathways in the graph and provide continuous communication to the base station. A proposed *O3DwLC* solution minimize the required number of relay nodes and links in the disjointed

pathways between adjacent nodes in the graph. The result evaluations conclude that a more reliable and robust WSN for monitored wildlife environment can be achieved.

An approach presented in [92], seeks to select a topology for mobile WSNs which improve the network connectivity considering the energy consumption among all of the nodes in the mobile network. The primary approach that the authors discuss in the paper is a clustered hierarchical topology which enables the possible leveraging of energy consumed among all the nodes in the network, in addition to allowing the network to reconfigure itself to prevent threats such as isolated nodes which may interfere with the overall network connectivity. The proposed solution consists of a hierarchical arrangement of the nodes in a clustered preferred topology and the use of a link metric parameter to assist the organization to measure the network connectivity. The outcome is a topology that can essentially reduce the unnecessary communication of nodes that are grouped into levels and clusters that restrict communication. The topology can deploy the nodes in different levels and connect them based on a predefined condition calculated by a proposed link metric. The defined parameter provides balanced energy consumption among nodes, which prolongs the network lifetime.

Often times, the use of the percolation theory for porous media is well suited for application in network models. A network model provides a representation of a porous medium that generally incorporates pore-scale descriptions of the medium and evaluates the physics of pore-scale events [93]. Network models and percolation theory are complementary to each other where network models have yielded insight into behavior at the pore scale. Additionally, the percolation theory has shed light on the natural effects of randomness in porous mediums.

Authors in [94], study the connectivity problem in WSNs for sandstorm monitoring giving consideration to the percolation theory. Sandstorm forecasting systems are deployed to serve different regions in sand rich middle eastern countries. The authors discuss distinct channel characteristics for four types of channels used in sandstorms monitoring and detec-

tion. The four channel types addressed include: air-to-air (AA) channel, air-to-sand (AS) channel, sand-to-air (SA) channel, and sand-to-sand (SS) channel. With the diverse sampling of channel protocols, the percolation-based theory is used to analyze the connectivity in shallow burial depths. The use of multiple types of channels is proposed to improve the performance of connectivity in the network. Accordingly, it is shown that smaller sensor density is sufficient to achieve the same connectivity performance when compared to the case of a single communication medium, i.e., terrestrial air channel. The studies above show the immense impact of connectivity reliability for environmental monitoring applications.

As presented in each of the sections above, extensive research have been conducted in the area of wireless sensor networks to provide techniques that can optimize sensor deployment, coverage and connectivity while minimizing the number of sensor nodes required to completely monitor the area [95] [21] [96]. However, the major focus of these papers only give consideration to the video sensor camera placement utilizing simple 2D sensing field coverage. In contrast, our model employs a pan-tilt directional D perspective, where the visual sensor has vertical and horizontal freedom. The ineffectiveness of these solutions for similar use in WVSNs is addressed below.

3.2 Wireless Video Sensor Networks

Over the past decade alone, WSNs have denominated many research areas by providing reliable and scalable technology. As a result there is now a growing interest in WVSNs [97]. There are considerable differences between WSNs and WVSNs that prevent the use of techniques that are already well developed for WSNs to be applied in WVSNs. Introducing WVSNs into an environment presents additional challenges that are not often attributed to WSNs such as the quality of coverage in WVSNs that depend on the orientation of the video sensor. Another differentiating aspect of WVSNs versus that of WSNs is the sensing range of sensor nodes which are a function of the sensor's field of view (FoV), aspect ratio and near/far fields. In this section, a discussion of related studies that explore the distinct

challenges of WVSNs is explored.

3.2.1 Deployment

Within this section, current studies relating to video sensor node deployment in WVSNs is discussed. The antagonistic relationship between the rigid requirements of visual data transmission and the constrained indoor placement restrictions of sensor networks warrants the need for an optimal deployment scheme in WVSNs. Several of these works evaluate various contexts of the aforementioned problem. In the paper featuring Chow et al. [98], the authors use a simple model to provide maximum angle coverage in a Visual Sensor Network to generate a minimum set of sensors to cover all objects of interest. Using a distributed algorithm they are able to achieve minimum cover with an image resolution constraint. In contrast, most WSN models employ simple 2D sensing fields, where the sensor has omni-directional freedom negating the impact in which the angular direction of the video sensor can affect the network. A novel approach to sensor deployment is considered in [99] where a Pan Tilt Zoom (PTZ) camera is used in a WVSN. The authors highlight how a PTZ WVSN differs from a traditional WVSN in that there are extended FoV coverages and semi-structured data source nodes that allow for irregular or incomplete data transmission. A PTZA heuristic is employed to account for the adjustment time the sensor requires to capture the visual data. The research discussed in [100] explores the deployment problem for WVSNs while considering the coverage of the monitored environment. A mathematical model is then used to define the problem of complete coverage using a greedy heuristic algorithm FoVIC where the objective is to cover the largest number of uncovered nodes within the area. However, both papers formulate models using a 2D approach (i.e. the latter emphasizing a mathematical model) and use quantitative analysis to evaluate their solutions.

While some works on the deployment problem isolate and focus only on the issue of video sensor placement, there are others who consider coverage in identifying optimal de-

ployment solutions. As seen in [55], a computational-geometry based approach is employed often times when optimization is desired in the deployment scheme. In the computational geometry based approach, geometry is used to evaluate the problem and define the algorithm. Using this approach, the algorithm is designed to efficiently solve the problem using data structures and basic geometric objects such as: line segments, points and polygons. A directional sensor network is implemented in [101] to maximize the area coverage of randomly deployed sensors using a greedy algorithm. The study in [102] provides a very detailed 3D practical model of the outlined problem for coverage rate optimization. Using a Greedy Iteration Scheduling based algorithm, their solution allows for overlapping of the sensing field for two nodes which can have greater overhead. In [103], the objective of the proposed work is essentially to provide target coverage for directed sensors with consideration to rotatable angles. Both use similar greedy approaches to solve the deployment problem.

The study in [104], investigates placement strategies of camera deployments to achieve the maximum amount of visibility in designing camera network arrangements. Fu et. al tackles the camera network deployment problem defined as “How to place the cameras in the appropriate places to maximize the coverage of the camera network under some constraints?”. The authors identify three types of constraints: 1) task constraints (e.g., continuous tracking and complete coverage) 2) camera constraints (e.g., camera network types and camera parameters) and 3) scene constraints (e.g., monitoring area configuration and candidate placement location restrictions). To solve the homogeneous camera network placement problem, the authors propose a binary Particle Swarm Optimization (PI-BPSO) algorithm. In an effort to suggest real world application, this study models the problem in a 3D space while the surveillance area is restricted to a 2D ground plane. The solution presented is first simulated and then incorporates real cameras for the experimental testing. The evaluation results show nominal improvement in the deployment numbers for sensors, however the 2D aspect of the monitored areas may limit its application in real-world scenarios.

3.2.2 Angular Coverage

Conventionally, sensor redundancy is considered to have negative impacts on the performance of the network. Authors in [105], argue that redundancy should depend on the nature of the monitoring applications. Essentially, within WVSNs the concept of redundancy relates to the coverage metric where the job of a sensor is to provide the required information for the application. As an example, coverage in WVSNs may call for sensors to cover equivalent (e.g. redundant point sets) areas ensuring full coverage and fault tolerance in the event of node failures.

The k -coverage problem generally states that any point within the monitored area must be located in the sensing range of at least k -sensors [106]. For example, if the deployment area is defined as 4-coverage, every point within the area is covered by at least four sensors. This means that up to three of the four nodes sensing the same monitored region can fail and the area will still be covered (monitored) by one of the four nodes. Full k -coverage denotes that every point is covered by all k -sensors. However, it is often suggested that guaranteeing 100% coverage in WVSNs is very difficult in a random deployment scenario. The authors in [56] employ a directional k -coverage metric (DKC) demonstrating the coverage quality measured in terms of a probability guarantee. Discussed below are schemes identified in related works addressing the traditional coverage problem for WVSNs.

The research conducted in [51] [107] [56], all provide detailed analysis of the coverage problem with consideration to 3D coverage in Costa et al., optimal angular placement in Yildiz et al. and directional k coverage in Liu et al. These solutions offer elegant 3D sensing models for the problem specified. However, when applied using real world constraints the solutions inadequacies are evident. Several studies have explored the issue of the coverage problem with some specifically addressing k -coverage. However, introducing video sensors into the narrative requires a different approach for optimal coverage.

It has been stated that video sensors collect data in a different way than sensors in WSNs. The sensing range of sensors in WSNs can be approximated and allow for an

omni-directional sensing model. However, for video sensors the sensing range in wireless video sensor networks equates to the FoV and is directional. Hence, the network concept of sensing uniformity is only valid for communication ranges (i.e., disk sensing model), which are omni-directional. A contrasting narrative is needed, where consideration is given to the angular direction for deployed visual sensors within the coverage area.

The work in [98] directly studies the angle coverage problem in visual sensor networks. In their work, the authors modeled the Minimum Cover problem and developed a distributed algorithm to determine the minimum set of sensors using an omni-directional sensing field. In [108], the authors address the k -coverage problem in dense sensor networks. The work formulated the k -coverage problem as an optimal hitting set problem. The proposed distributed algorithm in their work ensured k -coverage of the monitored area without requiring the location of the sensor node deployment. All papers highlighted above implement simple 2D models with 360° sensing fields within a WSN, resulting in impractical real world application settings and thus incurring inconsistencies. Another work implements the k -coverage problem as seen in [109], which addresses the problem in a three dimensional aspect but the sensing field range is modeled as an omni-directional sphere and the authors focus on solutions for wireless sensor networks which is inapplicable to the general coverage problem. As highlighted previously, introducing WVSNs into an environment presents additional challenges that are not often attributed to WSNs such as coverage quality that depends on the orientation of the visual sensor. In our work, we use the minimum number of visual sensors to achieve 2-angular-coverage from different directional perspectives to monitor 3D areas of interest.

3.2.3 Connectivity

Generally, many video monitoring applications require reliable continuous communications in scenarios where packet reconstruction of the original data is necessary. Moreover, some video sensors may transfer vital information for the application, where connectivity

interruptions can have devastating consequences. Therefore, the issue of connectivity in WVSNs is a critical factor in improving and maintaining high quality network performance.

Many studies evaluate the connectivity problem in addition to other factors (e.g. coverage, traffic, network lifetime) that affect WVSN performance. Though in WVSNs, one is not dependent upon the other and can in-fact be separate problems. When coupled with other existing factors in WVSNs such as the coverage problem, connectivity is often times only vaguely addressed or studies give assumptions for the connectivity metric [51] [42] [25]. As seen in several studies, implementation of a Wireless Local Area Network (WLAN) mesh system is used to construct multi-radio, multi-hop wireless mesh networks (WMNs). WMNs use multiple radios at each node location and provide multiple directional antennas to provide communication channels. The technology is applicable in commercial and government organizations for use in the deployment of WVSNs for surveillance in battle zones, professional sporting and concert events. The main design of the system is to fully exploit link layer characteristics to enhance configuration flexibility and network performance.

The study in [110] investigates the challenging issue of network bandwidth optimization for connectivity improvement. The authors propose a novel approach to multi-channel Wireless Mesh Network (WMN) architectures. The goal of the paper is to provide an effective solution to address the bandwidth problem by exploiting non-overlapping radio channels available through implementation of the IEEE 802.11 standards. In [111], authors highlight the current challenges of Visual Sensor Networks (VSNs). A current overview of the major research issues of VSNs is explored. Specifically, addressing coverage optimization, network architecture and power consumption for data communication. However, the issue of connectivity for VSNs is mentioned in passing when the authors address the coverage metric. This is a common occurrence of vague and ambiguous dialogue in the approach to connectivity. Interestingly, an approach for a separate WIFI framework is presented in [45], the work proposes a wireless video sensor network protocol for commercial and public safety. The proposed network platform develops a high-resolution (HS) surveillance system and wireless

networking platform with smart functioning parameters. The developed system can handle and capture high-resolution images. The images are then transmitted via a high speed wireless network (e.g., Wi-Fi Mesh). In an attempt to promote energy efficiency, the network system is divided into two categories: 1) a low-powered sleep wake cycle and 2) a high active cycle performance state. A master sub-system is comprised of an ARM Cortex A9 processor and implements a OpenWRT to manage the system. Then a slave sub-system is utilized to handle the topology management and allow for low-power maintenance of the entire system. The proposed system can be used for separate infrastructures in wireless video sensor networks.

3.2.4 Obstacle-Awareness

In real-world environments, such as home and office spaces, obstacles (e.g., furniture, lightning and miscellaneous decor items) are prevalent. These “*obstacles*” play an integral role in everyday life, facilitating daily chores, work assignments and providing ergonomical comfort. However, obstacles can potentially degrade the functionality of WWSN performance. Several studies have addressed obstacle detection for WSNs [112] [113].

Specifically, Wang et.al [112] proposes an estimation scheme for detecting obstacles in WSNs. The scheme identifies the obstacles using sensor node indicator markers around the obstacle boundaries. The scheme does not require the precise positional location for each individual nodes in the sensing field, however, a ratio function is implemented. The efficiency of the scheme then is evaluated using the network simulator ns-2.

A robot-deployment algorithm is proposed in [114] to handle obstacle detection in WSNs. The authors assert that the proposed study is to extend network lifetime and achieve full coverage by minimizing the number of required sensors. The robot-deployment algorithm consists of two states: 1) steady and 2) obstacle cycles. The concept of the solution is to perceive sensors that are deployed in the obstacle state as virtual obstacles. During the steady cycle, a spiral movement and node placement policy is given. When an obstacle is encoun-

tered, the obstacle cycle is activated. The algorithm then switches to the obstacle state, wherein the robot implements the surrounding movement policy to move and deploy sensors which reduce the impact of obstacles on deployment. The algorithm is evaluated using simulations. The authors in [115], proposed a computational geometry based approach for deterministic full coverage in WSNs with consideration to arbitrary boundaries and obstacle detection. An Optimal Regular Pattern Deployment (ORPD) scheme is used for plane covering within monitoring regions. The algorithm then determines the uncovered holes (e.g., subareas uncovered near obstacles and the region boundary) and places sensors for both regular and irregular obstacles. All of these studies use simple 2D models to formulate the deployment problem with consideration to obstacles, costing significant resource (i.e., sensor nodes) allocations and impractical solutions for WVSNs.

The majority of studies regarding multimedia sensor coverage consider an obstacle-free sensing environment [116] [117] [118] [119] [120]. All of the studies mentioned above address various performance metrics for WVSNs, where Munishwar et al. in [120] presents an overview of coverage algorithms, Costa et al. and Neishaboori et al. in [116; 118] focus specifically on target coverage respectively, Guo et al. [117] tackles area coverage and Yap et al. [119] evaluates both coverage and data transmission. None of the studies discussed above propose network models with consideration given to obstacles within the monitored environment, resulting in naive solution models and compatibility issues in real-world applications.

The study in [121] considers obstacle avoidance by finding the orientation for each of the video sensors that have rotational sensing abilities. The deployment scheme deploys a large number of low-cost multimedia sensors equipped with miniaturized cameras. The objective of the study is to minimize overlapping areas by exploiting FoVs converging regions in order to construct cover sets. A scheduling algorithm is proposed to ensure the maximum coverage of the areas between the video node and the subset of its neighbors.

Though, this study considers obstacles within its problem formulation model, the

proposed solution is limited by the incorporation of the following assumptions, where it assumes only three different rotational facing directions for the sensors and target regions within a two-dimensional plane model. Furthermore, this approach does not provide an optimal solution. However, our work addresses obstacle awareness with WVSNs where we can perceive and track obstacles within a 3D indoor space.

CHAPTER 4

OPTIMAL DEPLOYMENT

The chapter starts by introducing key terms and concepts in Section 4.1 that are prudent to understanding the formulation of the general deployment problem as it relates to WVSNs. An overview of key characteristics for 2D and 3D models is then provided to highlight the differences in each and to emphasize the trade-offs of using each model for the WWSN deployment problem in Section 4.2. Section 4.3 presents an in-depth discussion of the deployment problem where the problem is modeled using both continuous and discrete approaches. The proposed Greedy Heuristic and enhanced Depth First Search solutions are outlined in Section 4.4. Finally, an evaluation of both algorithms performance is presented in Section 4.5.

4.1 Introduction

In order to adequately grasp the intent of the identified deployment problem issue presented within this chapter, a basic understanding of core network jargon and sensor deployment concepts is required. Presented below are the definitions of commonly used terms mentioned throughout Chapter 4 and upcoming chapters.

- **2D** - is a term used for two-dimensional planar surfaces having only width and height (where both are in the same plane) but no depth aspect of positioning for a spatial environment (i.e., a building).
- **3D** - is an abbreviation used for three-dimensional spaces where both width, height and depth (where not all three are in the same plane) are considered for locational positioning denoted by axes x , y and z in the Cartesian coordinate system.

- **deployment** - is a planned or random approach to distribute sensors within a spatial environment (i.e., indoor or outdoor areas) considering constraints for detection, monitoring or surveillance applications.
- **depth first search** - is a traversal or search algorithm for tree and graph (graphs may contain cycles) data structures where initially a root node (i.e. start node) is selected arbitrarily from other node candidates and then explores as far as possible tracking and updating visited nodes along each branch before backtracking and only exits when the search node has been found or the structure has been fully explored.
- **greedy approach** - is an intuitive algorithm that selects the locally optimum choice in each iterative decision step in an attempt to find the globally optimum solution.
- **heuristic** - is an algorithmic paradigm or technique for solving common optimization problems in a feasible time frame using an approximate solution when traditional methods are not suitable.
- **relay node** - are a class of messenger nodes that may have longer battery life and computational power to provide communication ranges that exceeds that of other sensors within the network to facilitate the transfer of data.
- **WiFi** - a technology using radio waves that allow sensors and other devices to connect to the Internet or communicate with one another wirelessly within a particular area under the 802.11 standard for WLANs.

4.2 Model Comparison

The proliferation of WWSNs into virtually every aspect of our daily life has resulted in the exploration of methods to improve the technology as it relates to deployment and its sensing ability. In order to adequately study the deployment problem in WWSNs, a suitable model is required for proper representation of the environment to determine the position

and orientation within the monitored space. In existing literature, it is commonly assumed that the environment can be monitored in one of two ways, using either a 2D or 3D model. A presentation of the characteristics and limitations of each type of model is noted in Table 4.1 below.

Two-dimensional (2D) and three-dimensional (3D) models are a representation of actual dimensions corresponding to the physical structural environment that are mapped within a projected cognitized virtual workspace. The tradition of representing space as a 2D model theoretically and conceptually can be dated back to the early 1940s where researchers used them for operational research implementation [122]. In 2D models, a real world spatial environment is represented as a bi-dimensional space. The 2D geometric model provides planar projections of the physical monitored area. The two dimensions are commonly measured using length and width, where both directions must lie within the same plane. A sequence of n real numbers are adapted as a location in n -dimensional space. Thus, a two dimensional space can be denoted as n dimensional, where $n = 2$ and the set of all such locations is depicted as two-dimensional in a Euclidean space. The traditional characteristics of 2D models include representing the distances in the plane using Cartesian coordinates defined on x and y axis as shown in Figure 4.1a. Accordingly, Figure 4.1b illustrates sensor deployment within a flat planar space characteristic of most 2D network models.

However, in 3D models, real world spatial environments are represented as tri-dimensional spaces. Also within 3D environments, geometric models provide a graphical (i.e., perspective) projection of the physical monitored area. There are four parameters that can be adapted for 3D spaces, these include: height, width, depth and breadth. Three out of the four possible dimensions are commonly used for measurements in the space (i.e., using length, width and depth) where only two of the three directions can lie within the same plane. As stated previously, given a sequence of n real numbers adapted as a location in n -dimensional spaces. A three dimensional space can be formally defined as n dimensional, where $n = 3$ and the set of all such locations is depicted as a three-dimensional environment in an Euclidean space

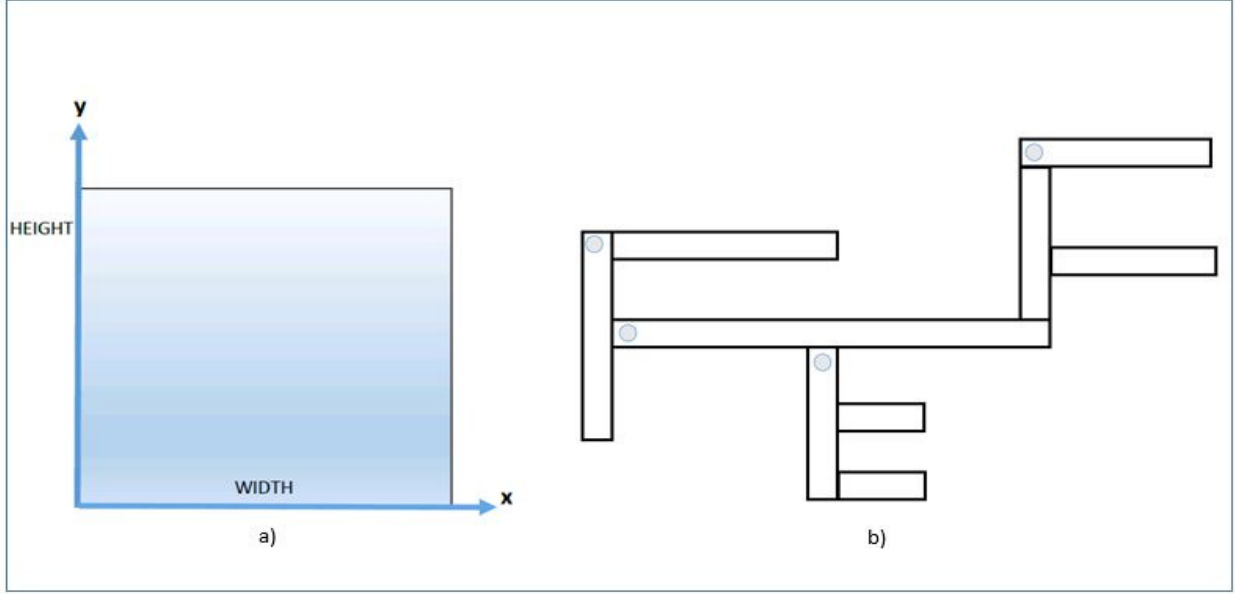


Figure 4.1. 2D Deployment Model

with coordinates x , y and z as presented in Figure 4.2.

After an extensive review of existing models utilized by numerous studies for WWSN deployment, our assessment revealed that many only assumed a very simple coverage model with no consideration to the orientation of the camera position (angular direction) [48]. In many instances, the studies lacked realistic models when defining the deployment problem and overlooked other performance metrics for the network deployment models[21].

Though optimization deployment techniques in 2D continuous mediums are well developed, it can be gleaned from the discussion above that there are in fact many challenges in directly utilizing 2D deployment methods for 3D indoor environments, where deployment area restrictions further render the solutions for traditional scalar sensors and 2D sensing fields incapable of solving the WWSN deployment problem for 3D indoor space monitoring. In the section below, the initial steps are presented to tackle this challenging deployment problem for WWSNs in 3D indoor space monitoring.

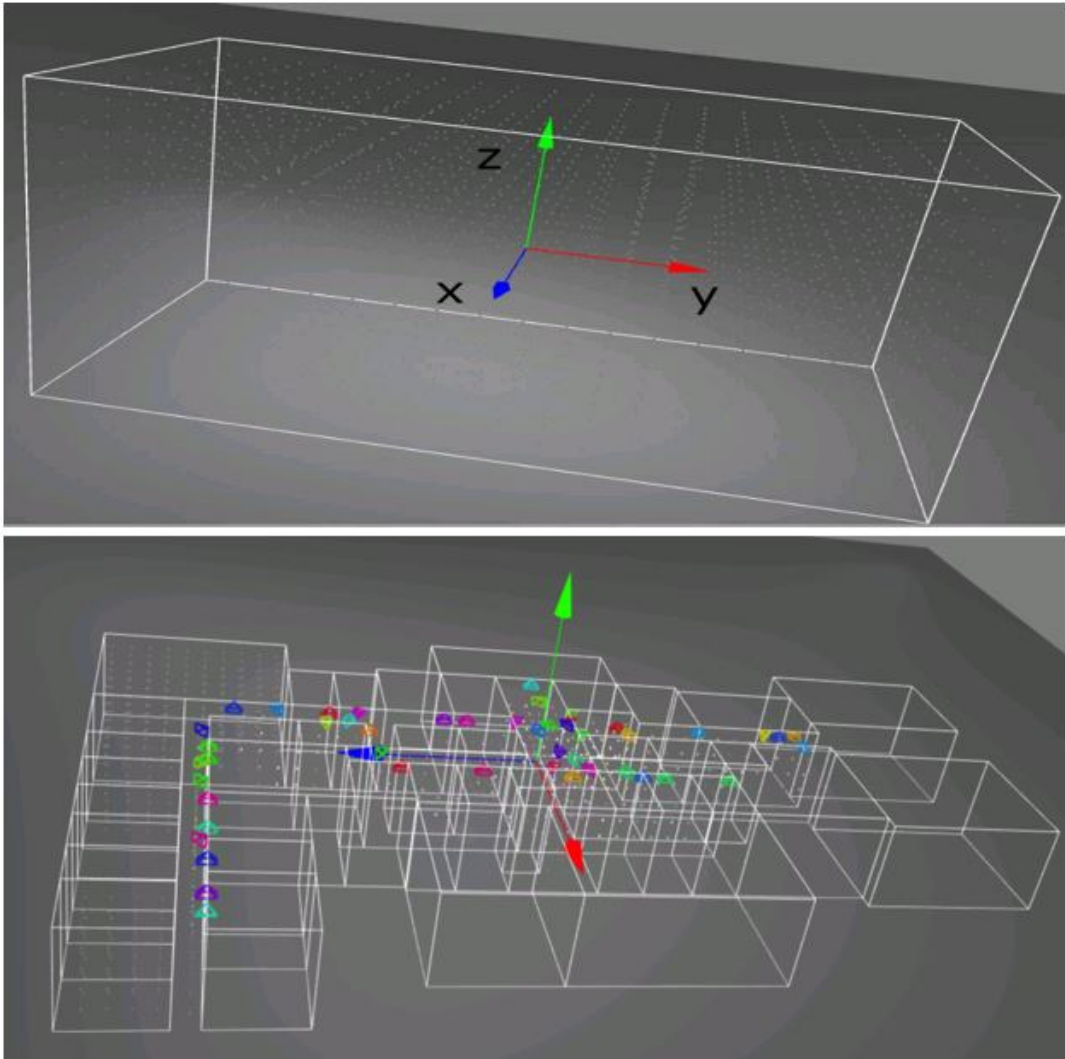


Figure 4.2. 3D Sensor Deployment Environment

Network Model Comparison		
Types	2D	3D
Aspects	Length and height	Length,width and depth
Data	Scalar	Visual
Representation	Flat planar	Graphical perspective
Coordinates	x,y	x,y,z
Examples	Rectangle, square, triangle	Cylinder, sphere, cube, pyramid, prism
Application	Health Monitoring, Home Temperature Monitoring	Security, Environmental Monitoring, Medical, Virtual Gaming
Advantages	Well studied, numerous implementation techniques, cost effective	Applicable to real-world scenarios, diverse in application use
Limitations	Incompatible with real world applications	Cost, requires continuous connectivity
Implementation	Simple	Complex

Table 4.1. Comparison of 2D and 3D Models

4.3 Problem Formulation

The section above highlighted some of the advantages in utilizing a 3D model to tackle the deployment problem for WVSNs in a 3D indoor space. The following section will outline the steps taken to solve the problem. We start by first modeling the general deployment problem in a continuous space (i.e. infinitely divisible space), where we implement techniques to minimize the number of required video sensors to cover the given 3D regions. We then address the problem by converting it into a discrete version (e., distinct set of values) where we incorporate 3D grids for our discrete model, which can achieve arbitrary approximation precision by adjusting the grid granularity. Consequently, by using the discrete model we can get very precise and realistic coverage space for each wireless video sensor. Next, we developed an enhanced Depth First Search algorithm that consists of an enhanced graph traversal method that searches the lattice of local candidate sites for optimal sensor node placement and angular direction. An area coverage function with a greedy heuristic, a derived *lower bound* for search branch pruning and a simulated frustum culling method are also utilized to increase the efficiency of the algorithm. A thorough analysis on the

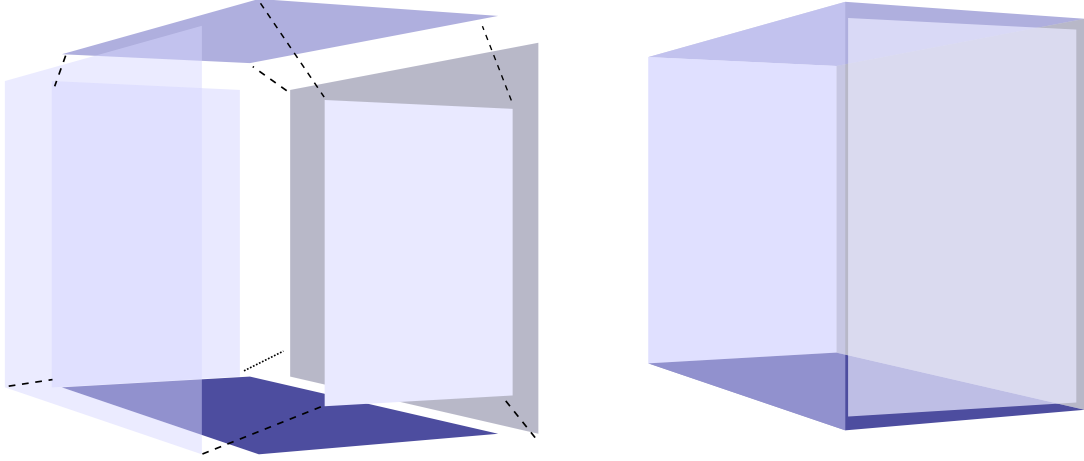


Figure 4.3. 3D Environment

experimental result follows.

The directional feature of WVSNs greatly impacts the deployment tactics implemented for coverage and other metrics as well. Essentially, coverage areas will vary due to the angular direction of the video sensor. Moreover, based on the hardware features of video sensors, the sensing range is often described as fan-shaped in 2D and cone or pyramid-shaped in 3D . Therefore, not only does the location of a video sensor affect its coverage area but the direction can affect the area it covers.

Hence, our network model considers a 3D space as illustrated in 4.3 (e.g., one floor area of a building), where some 3D regions (e.g., corridors) are required to be fully covered by a number of video sensors. The possible deployment locations are areas such as walls and ceilings that can be used to place video sensors. When a video sensor is deployed, its facing direction is adjusted to some certain position and then will not change during the whole monitoring period. Our goal is thus to optimize the placement and facing direction of each

video sensor, so as to minimize the number of video sensors required to fully cover all the required 3D regions.

Table 4.2. Parameter Notation Chart

Notation	Definition
\mathbf{S}	Video Sensor
\mathbf{A}	3D indoor monitored space
\mathbf{L}	Deployable location in \mathbf{A}
g_A	Granularity of grids in \mathbf{A}
g_L	Granularity of grids in \mathbf{L}
g_D	Granularity of facing sphere
(L, D)	location and direction of \mathbf{S}
$\mathbf{C}(L, D, R_S)$	Cover function
FOV	Video sensor field of view
F	Grid points in \mathbf{A}
F_C	Covered grid points in \mathbf{A}
C_L	Set of deployable location in \mathbf{A}
D	Direction of sensor
R_S	Sensing range
$best$	Locally optimum solution
$depth$	Search branch distance from root

4.3.1 Continuous Space Model

Given a 3D indoor area, let \mathbf{A} denote the 3D region that must be fully covered by the video sensors and \mathbf{L} denote the areas that can be used to deploy the video sensors. We define a tuple (L, D) to denote the location and direction of a video sensor. For a 3D space, the location L is represented by 3D coordinates (x, y, z) and the direction $D = (x', y', z')$ is a point on the surface of a unit sphere (which we call a facing direction sphere) with its radius equal to 1 and centered at $(0, 0, 0)$. We use *face* to denote that the facing direction is the vector from $(0, 0, 0)$ to (x', y', z') , as shown in Figure 4.4, where surface area points are mapped to the spatial sphere. Additionally, we use a to denote the spatial coordinates (interest areas) specifically within the 3D regions in \mathbf{A} that we want to cover. R_S is used to denote the maximum sensing range of the video sensor and we assume that it is the same for all the wireless video sensor nodes in the network. Let $\mathbf{C}(L, D, R_S)$ denote the

area that a video sensor can cover, which is thus a function of the location, direction and maximum sensing range of the video sensor. The video sensors in this study are modeled after a perspective camera as shown in Figure 4.5 which has static parameters: *farfield*, *nearfield*¹, *field of view* (FOV) and *aspect ratio*.

Thus, the problem is formulated as to find a set of locations and directions of video sensor nodes $S = \{(L_1, D_1), (L_2, D_2), \dots, (L_n, D_n)\}$, subject to the following constraints:

(1) Sensor Location Constraint:

$$\forall (L, D) \in S, L \in \mathbb{L} ;$$

(2) Area Coverage Constraint:

$$\forall a \in A, \exists (L, D) \in S, \text{ such that } a \in C(L, D, R_S) ;$$

Our objective is thus to minimize $|S| = n$.

4.3.2 Discrete Space Model

In discrete models data can be characterized as a countable set of values (i.e., integers) which are not infinitely divisible and can be empirically analyzed. Thus, for the discrete based approach we focused on two specific parameters within the 3D model: a set of candidate locations (positions) of the sensor nodes to identify the optimal location and the directional angle (orientation) of a video sensor node. In the continuous space model an infinite number of points exist, so we implemented a discrete lattice based grid model to approximate the continuous space model as in [42]. Specifically, we divided each region that must be fully

¹For ease of exposition, here we assume the *nearfield* is 1 and the *farfield* is R_S . A listing of important notations is provided in Table 4.2. As we will demonstrate in the discrete model, our model and solution can be easily adapted to other settings.

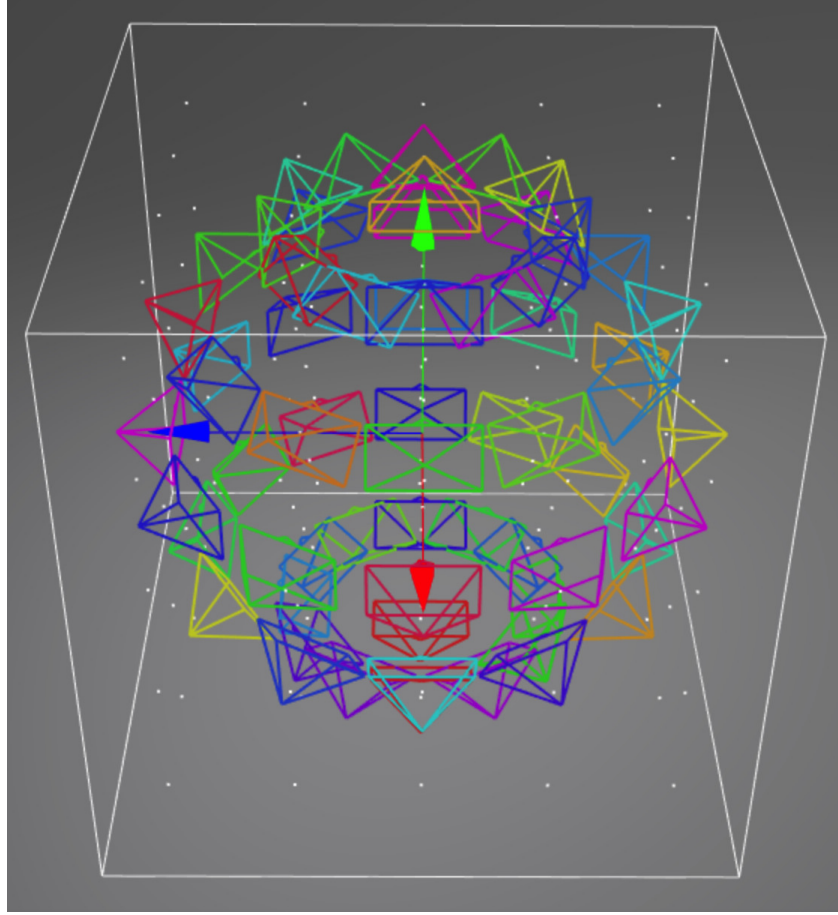


Figure 4.4. Facing direction sphere of video sensor with granularity of direction g_D

covered into discrete 3D grids as shown in Figure 4.6. The Bravais lattice structure was the inspiration for our model representing the 3D monitored environment. In geometry, a Bravais Lattice is a non-finite array of discrete points. The model is comprised of simple cubic unit cells which serve as the projected 3D Euclidean space. The simple cubic unit cell consists of 8 vertices, 12 edges and 6 planes. Given a 3D lattice cubic unit cell (i.e., the simplest repeating unit in a simple cubic structure) specifically the simple cubic (SC) with volume v^3 where v is the number of edges in one unit cell. We defined the length of v as one unit cell. Within \mathbf{A} there are *areas of interest* (subsections) denoted as $AOI = q(v^3)$ where q is the number of unit cells possible in a 3D space. Based on our modeling the ability to define AOI for specific target coverage is also available. The lattice points within the AOI that need

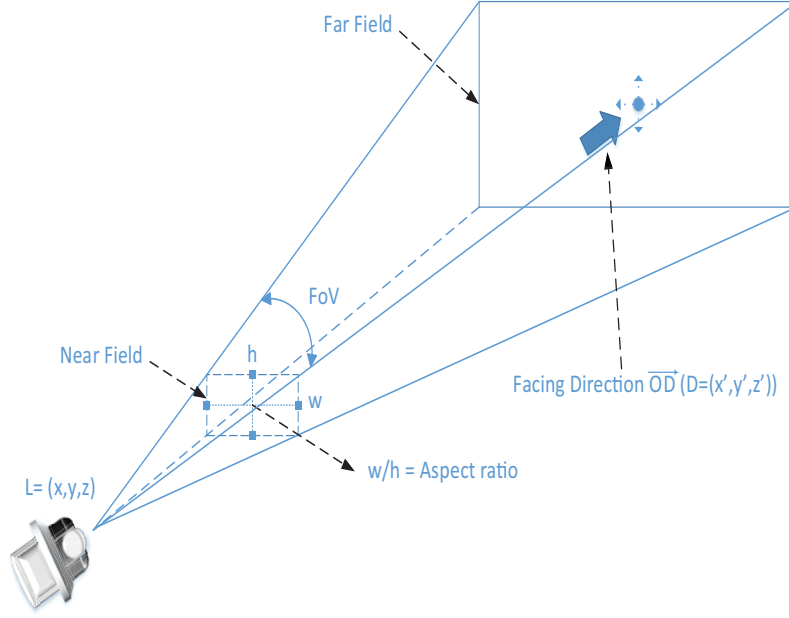


Figure 4.5. Video Sensor Parameters

to be covered by a video sensor are denoted by F (i.e. cover-able points within the grid). Additionally, we use F_C to denote the points that have been covered by a video sensor. A unit cell lattice point is completely covered if and only if the point that lies within the unit cell is covered by at least one video sensor. The set of possible location points for video sensor placement within \mathbf{L} is denoted as a discrete set $C_L = ((c_1, c_1, c_1), (c_2, c_2, c_2), \dots, (c_n, c_n, c_n))$ which is the set of *candidate locations* where a sensor can be attached (i.e., wall and ceiling locations considering the sensor location constraint). We utilized a similar method to divide each area that can be used to deploy the video sensors, where we assume that a node can only be deployed on a grid point within the area. The tuple (C_L, D) is used to denote the

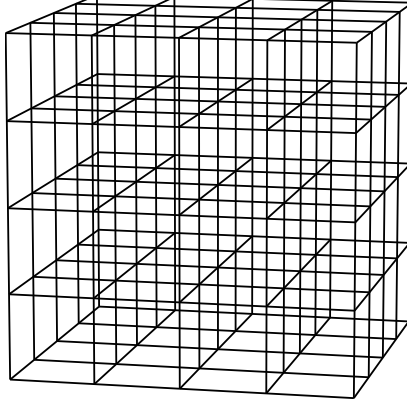


Figure 4.6. 3D Lattice Model

candidate location and direction of a video sensor. We use g_A to denote the granularity (i.e., the distance between two neighboring grid points) of the grids used in \mathbf{A} and g_L to denote the granularity of the grids used in \mathbf{L} . In addition, we also divide the surface of the facing direction sphere into grids (like the longitudes and latitudes divide the surface of the earth) and use g_D to denote the granularity. We assume that a wireless video sensor can only face to a direction where its D falls on a grid point.

4.4 Experimental Methodology

In the section below, we tackle the deployment problem by developing a greedy heuristic algorithm. We then proposed an enhanced DFS algorithm with pruning, which can yield high quality results efficiently and if given enough time can actually find the optimal solution.

4.4.1 Greedy Heuristic Algorithm

We now detail the design of our Greedy Heuristic algorithm where we can garner locally optimal candidate locations and then expand on this approach to improve our enhanced secondary solution. The objective of the greedy heuristic algorithm is to achieve complete area coverage of the 3D regions \mathbf{A} by determining the candidate locations C_L and directional angle D to cover the maximum number of lattice points, where each point is

Algorithm 1 Greedy Heuristic Algorithm

Input: \mathbf{A} , \mathbf{D} and \mathbf{L} **Output:** Set of S with max F in descending order*Initialize* : List of A , D , L and *final*.

```
1: while  $\mathbf{L} > 0$  and  $\mathbf{A} > 0$  do
2:   for each  $C_L$  in  $\mathbf{L}$  do
3:     if  $C_L \in$  locally optimal video sensor list then
4:       Compute all  $face \in D$  for  $C_L$ 
5:       Select  $face$  with max  $F$ .
6:       if  $C_L(F)^* > C_L(F)$  then
7:         Update  $C_L$ ;
8:       end if
9:     end if
10:  end for
11:  record  $C_L$  with max  $F$  into final
12:  remove  $C_L$  from  $\mathbf{L}$  and update  $\mathbf{A}$ 
13: end while
14: return final
```

covered by at most one video sensor. The detailed algorithm is presented in Algorithm 1. A while loop is implemented to check if there are coverable 3D regions and video sensor placement locations available to continue (line 1). Within the loop, we compute the *face* for all candidate locations C_L and sort the list of candidate locations based on the number of points that are covered (line 2-4). Instead of randomly choosing a location in \mathbf{L} to deploy a video sensor, in each iteration, the greedy heuristic algorithm strives to choose among the candidate locations in \mathbf{L} and find the location where the deployed video sensor can cover the maximum number of lattice points F (lines 5-7). Note that after a location is chosen by the greedy heuristic algorithm, since a video sensor is deployed at that location and covers a number of fresh points, the number of fresh points (i.e., points within the monitored area that have not been covered a video sensor yet) remaining to be covered by each remaining candidate location in \mathbf{L} needs to be recalculated (line 11). The one that maximizes the number of covered fresh points after recalculation will then be chosen as the next location to deploy a video sensor (line 12). A list of $S = (L, D)$ that will completely cover the 3D region is then returned (line 14).

4.4.2 Enhanced Depth First Search Algorithm

In the greedy algorithm, we exploited the inherent attribute of selecting local optimal camera coverings for the monitored area. Based on this, we use this number as our baseline for the enhanced DFS algorithm. The scheme of our enhanced DFS algorithm uses a traversal method to explore the branch paths within the deployment network whereby given enough time can improve the search performance. Traditional DFS algorithms often evaluate search branches redundantly resulting in an exponential run time because of expanded solution space in our case, considering the size of \mathbf{L} .

In a standard Depth First Search algorithm, we would need to explore each search branch that picks a location in \mathbf{L} and a facing direction. Since the solution space can expand quickly with the size of \mathbf{L} , this makes the algorithm very inefficient. In our design, we use pruning which can cut off most of the solution space and thus significantly improve the efficiency of our enhanced DFS algorithm, as presented in Algorithm 2. To achieve this, the first enhancement is that instead of starting the search from scratch, we use our greedy heuristic algorithm results as the currently found *best* solution², so that all the search branches (*depth*) that have already used equal or more number of video sensors compared to the currently found best solution can be safely pruned (lines 1-4).

The second enhancement is the selection of the location in each search step, instead of choosing the next location by the default order, we sort all the candidate locations by the decreasing order based on the maximum number of fresh points that a video sensor at these locations can cover and then choose by the sorted order (lines 10-14). This approach allows our algorithm to quickly find high quality solutions and skip as many low quality solutions as possible. Also, we apply a similar enhancement when we choose the facing direction of a video sensor. Another enhancement is that we derive a tight *lower bound* to estimate the number of video sensors that we still need to deploy to cover all the remaining fresh

²From the greedy algorithm, we calculate a number of cameras for covering. We use this number as our baseline for the enhanced DFS algorithm. At the first iteration, the “best” term equals to baseline.

Algorithm 2 Enhanced-DFS Algorithm

Input: \mathbf{A} , \mathbf{L} and \mathbf{D}

Output: Minimized set of $|S| = n$, where S is optimal

Initialize : List of A , D , L and *optimal*

```
1: DFS(depth)
2: if depth  $\geq$  best then
3:   return
4: end if
5: if  $A == 0$  then
6:   Update best;
7:   Record optimal;
8: else
9:   for each  $C_L$  in  $\mathbf{L}$  do
10:    Compute all face  $\in D$  for  $C_L$ 
11:    Select face with max  $F$ .
12:    Store  $C_L$  with max  $F$ .
13:   end for
14:   Sort  $\mathbf{L}$  by descending order of  $F$ ;
15:   Store  $C_{L^*} \rightarrow Queue$ ;
16:   while  $Queue \neq \emptyset$  do
17:      $C_{L^*} \leftarrow Dequeue$ ;
18:     if  $C_L \in$  final sensor  $S$  list then
19:       if ( $lowerBound(C_L(F)^*) + depth \geq optimal$ ) then
20:         break;
21:       end if
22:       Record  $C_L^*$  and  $F^*$ 
23:       Remove  $C_L^*$  from  $\mathbf{L}$  and  $F^*$  from  $\mathbf{A}$ ;
24:       DFS(depth + 1)
25:       Add  $F^*$  back to  $A$ ;
26:     end if
27:   end while
28:   Add all removed  $C_L^*$  back to  $\mathbf{L}$ 
29: end if
30: return optimal
```

points (lines 16-18). The *lower bound* is calculated based on the sorted candidate locations and directions in the previous two enhancements, where we keep choosing the location and direction from the front of the sorted results and add the number of fresh points covered by the chosen location and direction together until the sum is equal to or greater than the total number of fresh points that actually need to be covered (line 19). A recursive call to

the enhanced DFS search is implemented in line 24. When a search branch is cut off or fully explored the search will revert to its previous status (line 25-28). A minimized set of $|S| = n$, where S is optimal is returned as shown in line 30. The lower bound for the enhanced DFS algorithm is defined by the following equations:

$$\sum_{i=1}^n (C_L, D)^i_{F_c} \geq F_R, \text{ and } n + depth \geq best \quad (4.1)$$

where n is incremented by one until this constraint is satisfied. The number of video sensors (i.e., $(C_L, D)^i_{F_c}$) used to fully cover a region is tracked for 100 runs. We then use the smallest number of the total required to cover monitored region during this process as the lower bound. A lower bound is a metric to determine the minimum number of wireless video sensors required by the algorithm to cover the entire 3D indoor space (best case scenario). Given a subset S of some partially ordered set (D, \leq) , the lower bound is an element of D which is less than or equal to every element of S [123]. Dually, the upper bound is a metric to determine the maximum number of wireless video sensors required by an algorithm to cover the entire 3D indoor space (worst case scenario). Given a subset S of an ordered set (D, \geq) , it is an element of D which is greater than or equal to every element of S . In each search step, if this lower bound plus the number of video sensors that we have already deployed is equal to or greater than the currently found best solution, the search branch can be safely prune (line 4).

Theorem 1. *The enhanced DFS algorithm with pruning can return the optimal solution to solve the discrete version problem given enough time.*

Proof. The enhanced DFS algorithm incorporates three enhancements which improve the performance of the algorithm. The first enhanced approach that we apply is to prune some of the branches as the graph is traversed based on a greedy heuristic. Traditionally, a DFS algorithm searches a structure by selecting a root node and explores each branch then backtracks. A generalization of the traditional DFS algorithm is considered to be a brute

force approach where you search each branch until an optimal solution is returned. We can deem our solution to be a brute force approach with some of the infeasible solution space reduced. Thus, we will prove that our pruning feature in the enhanced DFS algorithm will not eliminate optimal solutions. We select and expand on search branches by determining the maximum number of fresh points covered by the chosen location and direction. In each iteration, we check how many new candidate locations n we need to cover the remaining fresh points F_R , which depends on the sorted order from the highest to lowest (i.e., best and worst case scenario) as a tight lower bound. We use the non-negative equation to check whether we need to prune it or not as considering the following formula: $n + depth \geq best$. If this non-negative equation is satisfied, we can cut the branch. Specifically, since the number of F_R is recalculated after each iteration of the search, the cost function is non-decreasing as the search step traverses the graph for a feasible solution. \square

4.5 Experimental Results

We conducted extensive simulations to evaluate our solutions using a customized simulator implemented by Java Script, which can emulate the 3D deployment of wireless video sensor nodes in a virtual environment as illustrated in Figure 4.7.

In our evaluation we used three algorithms: random, greedy heuristic and enhanced DFS. The enhanced DFS algorithm was allowed to run up to a time limit of 30 minutes to return the currently found best solution and it was successful in returning the optimal solution considering completion within the set time limit³ For our baseline approach we designed a random algorithm, which also served as a baseline to evaluate our solutions. In the random algorithm, we randomly chose a location in \mathbf{L} and deployed a video sensor at that position. We then adjusted the video sensor's facing direction so that it could cover a maximum number of fresh points in \mathbf{A} . After that, we continued to select another random location and deployed a video sensor there until all the grid points in \mathbf{A} were covered. It is

³A maximum time limit of 30 minutes is imposed on the enhanced DFS algorithm during the simulation phase to deem the solution feasible.

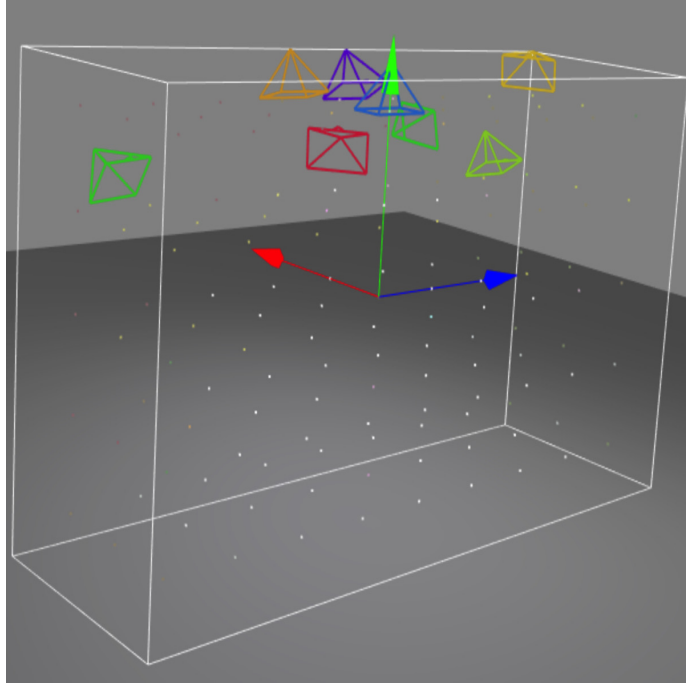


Figure 4.7. Network Simulation Environment

worth noting that during this process, if the video sensor at the randomly chosen location are unable to cover any new fresh points, we would remove the video sensor from that location and select another random location. Additionally, most deployment baseline methods presented in existing literature do not require heuristic methods to be implemented within random algorithms, for our presented approach, a heuristic is utilized and discussed above for enhancement during the comparison evaluations. The summarized standard settings for our simulations are displayed in Table: 4.3⁴. For each setting, we run the baseline algorithm 100 times and show the average of the results with an error bar to indicate the minimum and maximum values.

In our performance evaluations, the optimization of the video sensor deployment was considered where we looked at how varying the length for the 3D indoor space, candidate location positions, field of view of the sensor, near fields, far fields, granularity of the monitoring area g_A , g_L (candidate locations), and g_D (direction of S) in the space would affect

⁴The asterisk* within Table: 4.3 denotes that there are no abbreviated terms used for the parameters.

Table 4.3: Default Parameters for Simulation Execution

Parameter	Denotation	Default Value	Dynamic (D) or Static (S)?
3D Indoor Space	L	Length X 60 X 100	D
Monitored Area	A	Same as 3D Indoor Space	D
Deployment Area	Tuple(L,D)	Top half of walls and ceiling	D
Granularity of \mathbf{A}	g_A	20	D
Granularity of \mathbf{L}	g_L	25	D
Granularity of \mathbf{D}	g_D	45°	D
Field of View	FOV	50°	D
Max Sensing Range	R_{max}	100	D
Aspect Ratio	*	1.778	S
Near Field	*	1	S
Far Field	*	100	S

Table 4.3. Deployment Problem: Simulation Parameters

the performance of all three algorithms. We discuss some of the more interesting findings discovered in the testing using the figures below.

Depicted in Figure 4.8 are the results from our evaluation analyzing the impact of varying space lengths within the indoor monitored environment (which can be deemed as the corridors of various length). The greedy heuristic algorithm out performed the random algorithm for each length interval increase (in requiring a smaller number of video sensors to cover the monitored area) and the enhanced DFS required less video sensors compared to the baseline and greedy approach for two testing interval cycles, 250 and 300, respectively. There is a comparable reduction in the number of video sensors for both the greedy heuristic and enhanced-DFS algorithm. This can be attributed to the time limit restriction placed on the enhanced DFS.

When varying the granularity of the monitored points A (distance between coverable points) within the 3D region as shown in Figure 4.9 there is better performance for the enhanced-DFS which required less sensors to fully monitor the indoor space.

The granularity of the grids in D and L are evaluated in Figure 4.10 and Figure 4.11, respectively. The performance of the enhanced DFS is stable and continues to reduce the

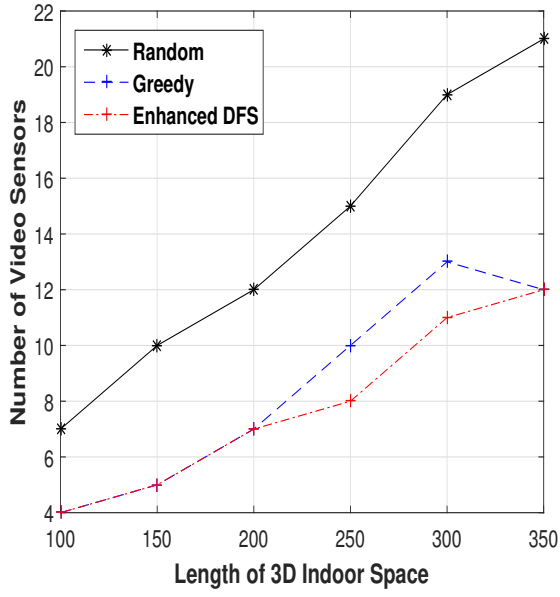


Figure 4.8. Dimensional 3D Indoor Space

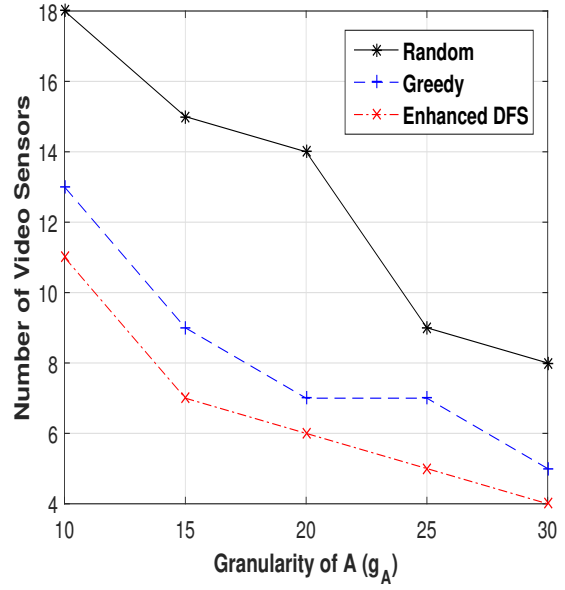


Figure 4.9. Varying g_A impact

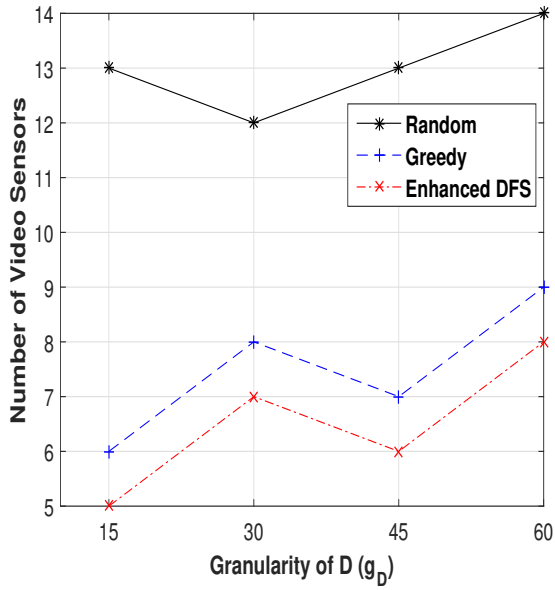


Figure 4.10. Varying granularity for g_D

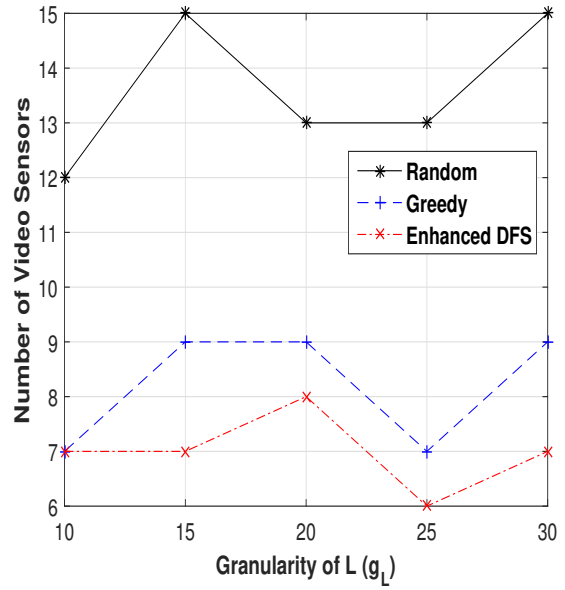


Figure 4.11. Varying granularity for g_L

amount of S for optimal coverage in the 3D regions. The performance testing for the near field (shown in Figure 4.12 variations resulted in a 50% reduction of video sensors compared to the random algorithm and it also fared better than the greedy heuristic algorithm.

Figure 4.13 showcases similar results for the reduction of required video sensors to

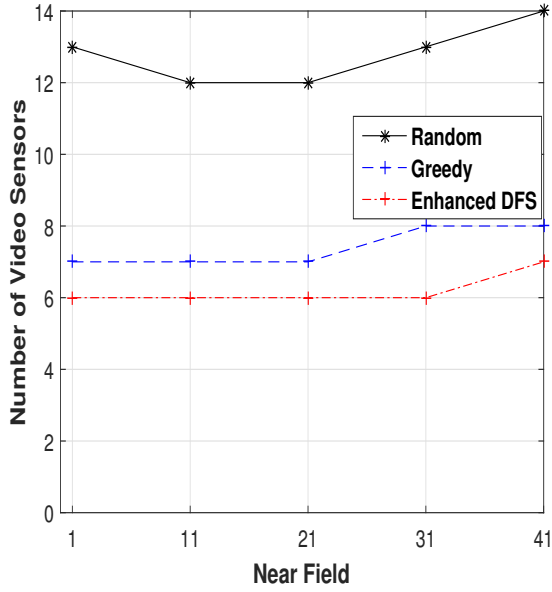


Figure 4.12. Impact of varying near fields

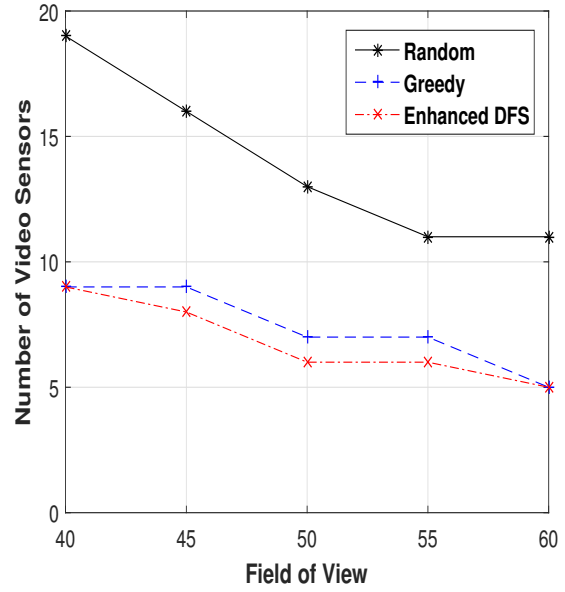


Figure 4.13. Varying field of views (FoVs)

cover the space when varying the FOV for S . The overall performance of the enhanced DFS is stable and continues to reduce the amount of S for optimal coverage in the 3D regions. The enhanced DFS algorithm can further reduce the number of video sensors by up to 20% over our greedy heuristic algorithm. In Figure 4.14, the performance for far field variations are explored. The enhanced DFS requires 50% less S than the random algorithm and performs better than the greedy heuristic algorithm.

For Figure 4.15, we evaluated the different domain variations of candidate locations (i.e limit the deployable area to smaller sections of the wall or ceiling). The enhanced DFS used less S to fully cover the area. After evaluating the initial results, it is clear to see that both our greedy heuristic algorithm and enhanced DFS algorithm outperform the random algorithm. In particular, compared to the random algorithm, the number of required video sensors can be reduced up to 50% by our greedy heuristic algorithm, and our enhanced DFS algorithm can further reduce the number of video sensors by up to 20% over our greedy heuristic algorithm. Based solely on these encouraging results, we continue on to investigate the coverage problem for WVSNs.

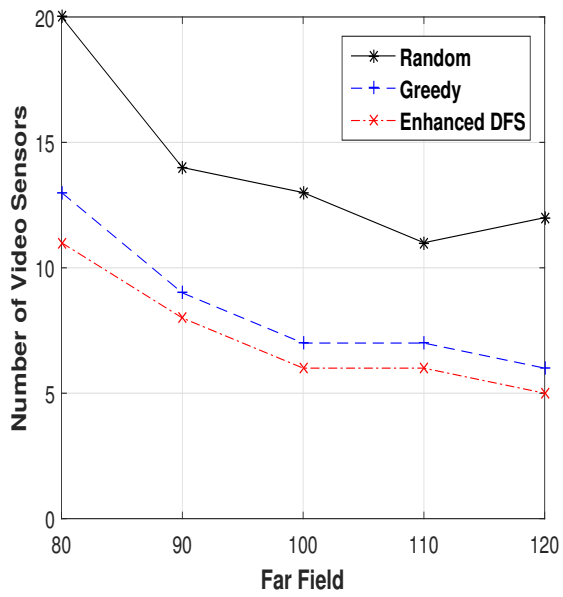


Figure 4.14. Impact of varying far fields

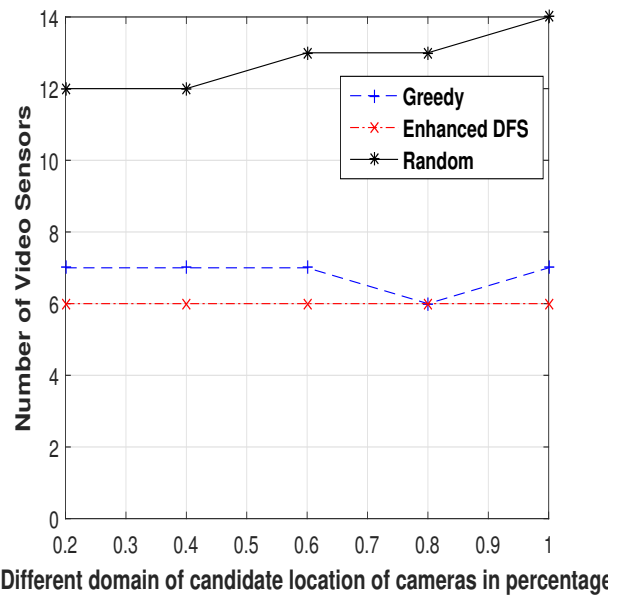


Figure 4.15. Impact of candidate domain size

CHAPTER 5

CONNECTIVITY AND OBSTACLE-AWARENESS

Extending the theme from Chapter 4, where we proposed schemes to optimize the performance of a fundamental issue in WVSNs by addressing the deployment metric, we now extend our focus to additional issues as it relates to connectivity and obstacle awareness in WVSNs. In Chapter 5, we address these new challenges by ensuring connectivity and providing mechanisms to handle obstacles within the 3D indoor environment. The chapter begins with an introduction on the motivation and key challenges in reference to the connectivity and obstacle awareness problem for WVSNs, in addition to defining key terms for the chapter in Section 5.1. Section 5.2, provides a brief discussion on the intrinsic properties of our model in comparison to more traditional models from existing literature. Next, we present the revised network model and formulate the problem considering additional constraints in Section 5.3. Section 5.4 presents an in-depth discussion of the connectivity and obstacle awareness problem where the proposed Greedy Heuristic and enhanced Depth First Search algorithms are given. Section 5.5 provides the performance evaluation for both algorithms.

5.1 Introduction

Connectivity is an essential asset within WVSNs, as it facilitates communication among the wireless sensor nodes. Connectivity provides “pathways” for each video sensor to not only communicate but also transfer data packets between each “connected” node to a sink (i.e., base station). A system network that fails to provide communication is greatly limited in its functionality. Also, when incorporating video sensor surveillance into a normal WIFI network, one has to consider the large amount of data traffic collected by the video sensors. This may cause the WVSN as well as normal users of the WIFI network to encounter

interference or interruptions in communication. Therefore, the WWSN is often required to have its own separate communication network to connect the entire monitoring system [124]. Thus, ensuring optimal connectivity is a valid and core issue in WWSNs, however, the solution is less than straight forward.

Another pressing issue in WWSNs, is obstacle awareness. Obstacles are a constant phenomena, especially in indoor 3D environments. Such factors have a direct correlation with an indoor environment setting (e.g., there are often obstacles such as ceiling lamps and furniture inside the indoor space), which, if not carefully considered, can easily block the line-of-sight of deployed video sensors reducing their sensing capabilities and thus the overall performance of the WWSN.

Therefore, not only are connectivity and the quality of the video sensor sensing ability affected by the communication range of a video sensor but also obstacle awareness, as well. Though there are recent studies available on the connectivity and obstacle detection metrics in WWSNs, however, they are considered separately instead of jointly [125; 126; 127; 128; 129]. The aforementioned issues highlighted above provides the motivation behind our exploration of the connectivity and obstacle awareness problem in this chapter.

Throughout Chapter 5, we will be using additional terminology to precisely describe our revised network model and formulate our problem. Presented below are the definitions of commonly used terms mentioned throughout Chapter 5.

- **connectivity** - is the ability of a network to establish communications pathways for every pair of nodes to allow the transferring of data among each sensor.
- **obstacle awareness** - refers to the capability of our algorithm to perceive or have knowledge of a object and the ability to avoid such points which are occluded by the object.
- **obstacle** - is a three dimensional object modeled within the network using predefined parameters to represent a more realistic environment.

- **base station** - refers to one or more sink nodes that have additional energy, computational and communication resources within the network that serves as a interface gateway capable of forwarding data between the video sensor nodes and the end user.
- **intersection point** - denotes the location where two intersecting lines cross paths (i.e., point of intersection), the term is used as a parameter in our obstacle detection strategy.
- **line of sight** - this term is associated with the visibility of the video sensor's field of view as it relates to the view frustum within the network.
- **communication pathway** - is an established link between a set of nodes beginning with a root node within the network to a base station, implemented using a model paradigm.
- **view frustum** - is the perceived volume in the three-dimensional indoor space that is visible to the video sensor, modeled as a perspective camera (i.e, refer to Figure 4.5).
- **segment** - this term refers to the line segment between the location of a video sensor and a specific monitoring point, whereby it is bounded by two distinct end points, containing the set P which is every point on the line between its two endpoints.

5.2 Model Comparison

The two types of models prevalent in existing literature as it relates to network connectivity include: deterministic and probabilistic (i.e., stochastic) paradigms.

Deterministic models require the use of tangible, factual data. In terms of a network model it corresponds to the communication model (i.e, communication range for the video sensor). In traditional models, a deterministic geometric disk model is used where nodes are connected if they are within a certain communication range from each other. The reason behind this model's popularity is its ability to easily determine if the data is true or false,

in our case the approach can determine if a sensor is within communication range of a neighboring sensor or not.

For example, in a network model where sensors are randomly deployment within the network, when implementing this paradigm two nodes are considered to be connected and form a pair if they are within a certain distance (i.e., communication range C_r) from each other [130]. This model can easily determine if the nodes are connected or not based on the C_r parameter.

Probabilistic models are meant to give a relative distribution of possible outcomes. This model is more complex in the way it determines the communication of a network, where it relies heavily on probability. The data is generated through collecting data points measurements from the probabilistic disk model and comparing it to the deterministic data points. Probabilistic communication models express the quality of communication between nodes by comparing it with a known connection pattern with similar behavior [131]. The likelihood of the measurement is given by probabilistic comparison (i.e., changes over time described by past patterns in addition to probabilities for successive change that are semi-predictable) of the actual communication paths within the expected sampled measurements [132]. An advantage of probabilistic models is its scalability.

In this dissertation, we implement the deterministic approach as a circular disk model whereby it is considered to be a stable approach, sufficient enough when modeling distance-dependent wireless pathways.

5.3 Problem Formulation

In this section, we build on the knowledge gained from Chapter 4 where we exclusively considered the deployment problem. In an attempt to closely model the reality of a real world environment within our 3D indoor space model, we reformulate our problem to consider both connectivity and obstacle awareness simultaneously.

The objectives for the connectivity and obstacle awareness problem is to provide qual-

ity sensor connectivity that is resilient to building communication infrastructure failures and can detect obstacles (i.e., to address the line-of-sight blockage of a video sensor) within the monitored space. The introduction of these additional metrics into our initial problem where the objective is to 1) optimize the placement and facing direction of each video sensor and 2) minimize the number of video sensors required to fully cover all the required 3D regions, greatly increases the complexity of the problem. To this extent, we explicitly developed both a connectivity constraint so that each deployed sensor has a path, composed of connected sensors, to reach the base station (i.e., sink) and an obstacle awareness constraint that improves the efficiency of the network by removing grid points (i.e., reduce search space) from the search space that are blocked by obstacles within the monitored space. Traditionally, this problem is investigated via utilizing 2D grids. However, our problem is considered in a 3D space.

5.3.1 Continuous Space Model

The continuous space model is extended from the prior chapter to include additional parameters for the inclusion of the connectivity and obstacle constraints. We consider a 3D indoor space, where some 3D regions (i.e., \mathbf{A}) are required to be fully covered by a number of video sensors. Also, there are some areas such as walls and ceilings that can be used to deploy video sensors (i.e., \mathbf{L}). When a video sensor is deployed, its facing direction is also adjusted to some certain position and then does not change anymore. Also within the indoor environment, there are often obstacles which introduce additional challenges as obstacles can block the line-of-sight of video sensors and reduce their sensing capability. Our goal is thus to optimize the placement and facing direction of each video sensor with consideration of obstacles and connectivity, so as to minimize the number of video sensors required to fully cover all the required 3D regions. The table below provides a list of important notations for reference (see Table 5.1).

We assume R_{max} is the maximum communication range achievable for a video sensor.

Table 5.1. Parameter Notation Chart II

Notation	Definition
S	Video Sensor
A	3D indoor monitored space
L	Deployable location in A
L_O	Base station location A
g_A	Granularity of grids in A
g_L	Granularity of grids in L
g_D	Granularity of facing sphere
(L, D)	location and direction of S
$\mathbf{C}(L, D, R_S)$	Cover function
A_i	Area of interest in A
F	Grid points in A
F_C	Covered grid points in A
O	Set of obstacles
$Path$	Communication path from (L, D) to L_O
O	Single obstacle in O
R_{max}	Communication range of S
$SegmentL_a$	Line segment between L and a
a	Specific monitored grid point

We use **O** to denote a set of obstacles that may block the line of sight for video sensors and $Segment(La)$ to denote the line segment between the location L of a video sensor and a specific monitoring point $a \in \mathbf{A}$.

Our general problem thus can be formulated as to find a set of locations and directions of video sensor nodes $S = \{(L_1, D_1), (L_2, D_2), \dots, (L_n, D_n)\}$, subject to the following constraints:

(1) Sensor Location Constraint:

$$\forall (L, D) \in S, L \in \mathbf{L} ;$$

(2) Area Coverage Constraint:

$$\begin{aligned} \forall a \in \mathbf{A}, \exists (L, D) \in S, \text{ such that} \\ a \in \text{Cover}(L, D, R_S) \text{ and} \\ \forall O \in \mathbf{O}, \text{Segment}(La) \cap O = \emptyset ; \end{aligned}$$

(3) Network Connectivity Constraint:

$$\begin{aligned} \forall (L, D) \in S ; \\ \exists \text{Path} = \{(L_{p_1}, D_{p_1}), (L_{p_2}, D_{p_2}), \dots, (L_{p_k}, D_{p_k})\}, \\ \text{such that } \text{Path} \subseteq S, L_{p_1} = L, \\ |L_{p_i} L_{p_{i+1}}| \leq R_{\max} \text{ for } i = 1 \dots (|\text{Path}| - 1) ; \\ |L_{p_k} L_0| \leq R_{\max} ; \end{aligned}$$

where $\text{Path} = \{(L_{p_1}, D_{p_1}), (L_{p_2}, D_{p_2}), \dots, (L_{p_k}, D_{p_k})\}$ denotes the communication path from a video sensor deployed at (L, D) to the base station at L_0 . Our objective is thus to minimize $|S| = n$.

5.3.2 Discrete Space Model

As stated previously, the monitored region \mathbf{A} is divided into grid points to approximate the continuous model as illustrated in Figure 5.1. As long as all the grid points in the region are covered, we consider the region is fully covered. The *areas of interest* within \mathbf{A} is denoted as A_i . For a more precise model, we further define lattice points within A_i denoted as F to be the remaining coverable points within the discrete grid model, F_c is denoted as the points that are covered. Similar symmetric division of the grid points is also employed to \mathbf{L} , where video sensors can be deployed at *candidate locations*, defined as $C_L = ((c_{x_1}, c_{y_1}, c_{z_1}), (c_{x_2}, c_{y_2}, c_{z_2}), \dots, (c_{x_m}, c_{y_m}, c_{z_m}))$. We assume that a video sensor can only be deployed on a grid point within \mathbf{L} . To allow more flexibility within our discrete model,

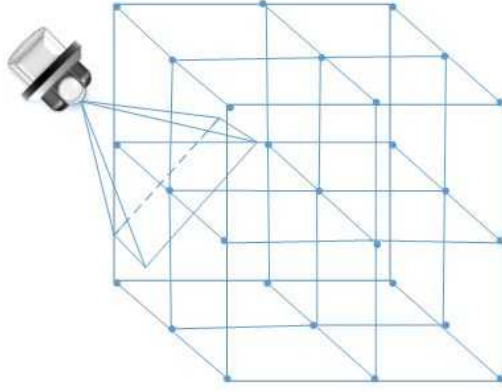


Figure 5.1. 3D Lattice Grid Model

the neighboring grid point distances can be adjusted using granularity denoted as g_A for \mathbf{A} . Similarly, g_L is used to denote the granularity of the grids used in \mathbf{L} . The discrete spherical model referenced in Chapter 4 is used (similar to the longitude and latitude coordinates used on the Earth's surface) to address the facing direction D 's granularity denoted by g_D , (see Figure 4.4.)

5.3.3 Strategies for Obstacle Detection and Connectivity

Real world indoor settings include furniture, ceiling lights, as well as other types of obstacles. So our next objective is to provide a strategy to handle this scenario in our discrete model. We consider the scenario where some obstacles exist inside the indoor 3D space and can obstruct the line-of-sight of wireless video sensors. For example, furniture, lighting and other decor are staples for indoor environments. In this dissertation, we focus on stationary obstacles as seen in Figure 5.2. When we embed wireless video sensors to cover the monitored space with existing stationary obstacles, we need to design several solutions to detect these obstacles and then make sure whether grid points near the obstacles are obstructed or not when being considered for coverage during deployment of a video sensor. The obstacle constraint within the area coverage constraint denotes that a monitored point

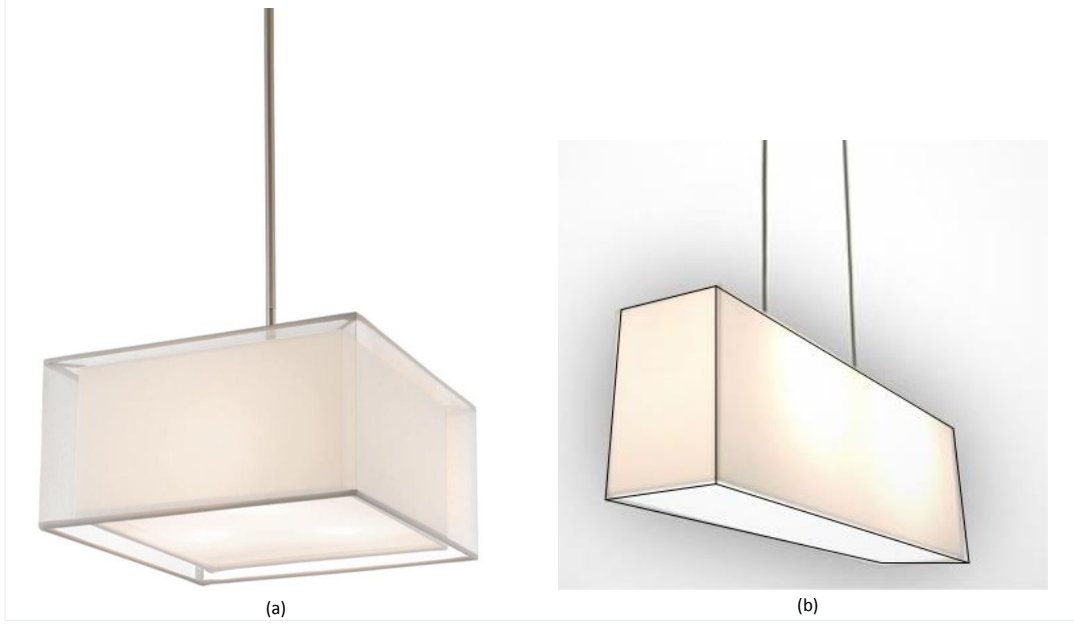


Figure 5.2. Representation of Lamp Obstacle (a) Actual Purchasable Lamp (b) Cuboid Lamp representation

a is within the covered frustum of $Cover(L, D, R_S)$, however it is not in the shaded area of any obstacle, which is denoted as $Shadow(C_L, O)$ and $Segment(L_a)$ does not intersect with any obstacle $O \in \mathbf{O}$.

The Divide and Conquer detection strategy allows for an efficient but incomplete detection of the obstacles and Accurate detection produces complete detection of the obstacle within the 3D space. We define the single obstacle $O(x_l, x_h, y_l, y_h, z_l, z_h)$ where “ x_l, y_l, z_l ” and “ x_h, y_h, z_h ” represent the low boundary and high boundary in X axis, Y axis, and Z axis for each obstacle, respectively.¹ For the Divide and Conquer detection we consider two

¹ For the obstacles with irregular shapes, multiple smaller cubes can be used to approximate the obstacle with arbitrary accuracy by adjusting the cube size.

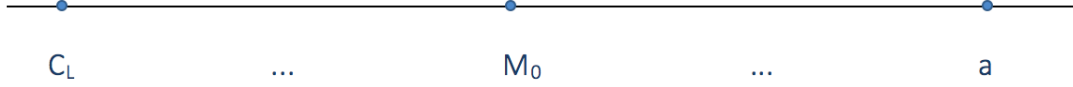


Figure 5.3. Representation for Divide and Conquer Detection strategy

parameters: C_L the location of a deployed video sensor and a monitored particle a in \mathbf{A} as illustrated in Figure 5.3.

Since we know point C_L and point a , we can check the middle point M_0 between C_L and a , to test if it resides within an obstacle, if yes the algorithm exits. Otherwise, we divide the $Segment(C_L a)$ into two sub-segments and then check the middle points of these two sub-segments. If any one of the two middle points is in an obstacle, then the algorithm stops. Otherwise, the algorithm will continue to divide and conquer, until we have completed this step n times where no middle points are located in any obstacles. For the size of n , we have to choose it wisely since there is a trade-off between efficiency and accuracy for the program. For an example, if the size of n is too big, increasing the detecting accuracy can lead to an increase in the overhead of the program performance. However, if the size of n is too small, the results may also lead to an inaccurate detection (missed obstacle detection) of smaller size obstacles in the monitored area.

The Accurate Detection strategy is a complete detection method which uses the points C_L and a to calculate the straight line equation $Segment(C_L a)$ in 3D space based on coordinates of these two points as seen in Figure 5.4. Using $Segment(C_L a)$ and obstacle $O \in \mathbf{O}$ we check whether there exist an intersection point between C_L and a . If an intersection point exists we can conclude that the monitored point a is obstructed by the obstacle within the 3D indoor space. The calculation procedure is shown below. Assume the coordinate of $C_L = (x_1, y_1, z_1)$, and the coordinate of $a = (x_2, y_2, z_2)$, we can get $Segment(C_L a)$ as:

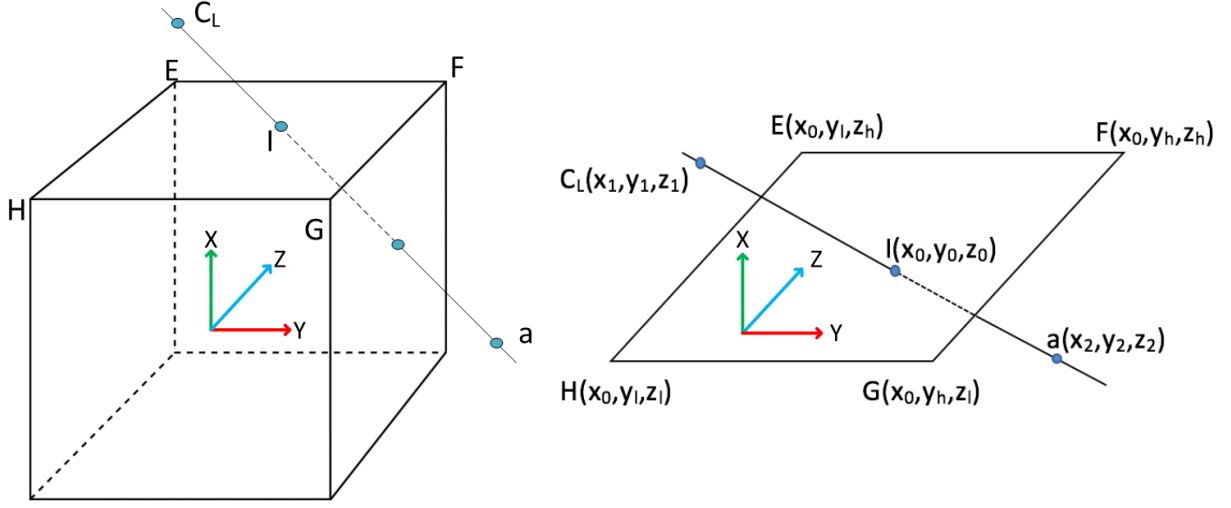


Figure 5.4. Accurate Detetection Model

$$\begin{cases} (x - x_1) = t(x_2 - x_1) \\ (y - y_1) = t(y_2 - y_1) \\ (z - z_1) = t(z_2 - z_1) \end{cases} \quad (5.1)$$

where t is an intermediate value, and (x, y, z) denotes any point on the straight line (C_L, a) . We use a cuboid shape to represent the obstacle, so each obstacle contains six surfaces. Next, we examine that if the straight line $Segment(C_L a)$ intersects with one of these six planes of each single obstacle, i.e., if $Segment(C_L a) \cap O \neq \emptyset$, the monitoring point is obstructed by the obstacle. Since the obstacle is cuboid, each plane is actually a rectangular. To simplify the problem, we assume that all the cuboids used to represent the obstacles are formal, which means that there exists an unchanged axis in the surface coordinate. In other words, all the points on this plane have one same axis value.

In Figure 5.4, we illustrate an example where the X axis dimension is fixed for surface

$EFGH$ with value x_0 , we can then calculate the value of y_0 and z_0 as:

$$\begin{cases} y_0 = (y_2 - y_1)(x_0 - x_1)/(x_2 - x_1) + y_1 \\ z_0 = (z_2 - z_1)(x_0 - x_1)/(x_2 - x_1) + z_1 \end{cases} \quad (5.2)$$

where $x_2 \neq x_1$. The constraint below is also implemented to test whether the line segment intersects with the plane as:

$$\begin{cases} x_1 \geq x_0 \geq x_2, x_1 \geq x_2 \\ y_h \geq y_0 \geq y_l, y_h \geq y_l \\ z_h \geq z_0 \geq z_l, z_h \geq z_l \end{cases} \quad (5.3)$$

Only if these constraints are satisfied, then the straight line $Segment(C_La)$ intersects with plane $EFGH$ at point I , which means that point a is obstructed by the obstacle. When $x_2 = x_1$, either the straight line $Segment(C_La)$ is in parallel with plane $EFGH$ (if $x_0 \neq x_1$) or part of the straight line $Segment(C_La)$ is inside plane $EFGH$ (if $x_0 = x_1$), where the latter case also indicates that point a is obstructed by the obstacle. This strategy guarantees that our result is correct. Figure 5.5 provides a representation of actual model obstacles in the 3D setting. One drawback is this strategy is slightly slower than the Divide and Conquer Detection strategy (i.e., where we divide the line segment in sub-segments at its midpoints) since more calculations are needed to accomplish it. We can use the accurate strategy in a scenario to check when the size of an obstacle is very small i.e, less than the granularities defined in the scenario.

Our next step for the proposed problem is to establish connectivity for the deployed sensor nodes in the network. We outlined our requirements in our network connectivity constraint. Traditional connectivity constraints employ paradigms whereby a connected communication path is formed by any number of arbitrarily paired of sensor nodes to establish connectivity. However, this constraint is too broad and tedious to implement. In our model,

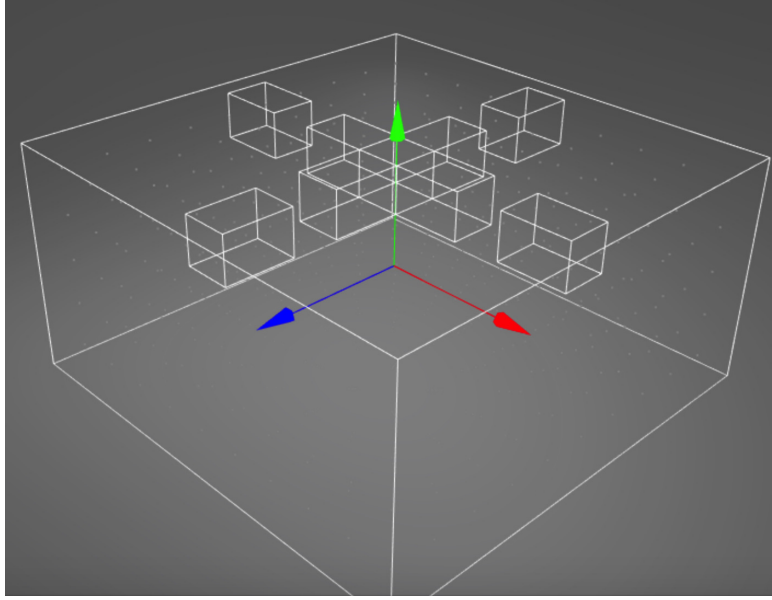


Figure 5.5. Simulated Indoor Obstacles

the approach we use employs a practical and realistic scheme, where instead of establishing a communication path between all of the nodes deployed, we determine a path to the base station (i.e., sink), this method is sufficient enough to ensure connectivity. A second layer of reliable connection can be established by utilizing a relay node to connect to a sink, however this approach was not implemented in our model.

5.4 Experimental Methodology

In the subsections below we discuss the algorithms used to tackle our problem. The main ideas of the algorithms used in the methodology herein are similar to those discussed in Chapter 4, however, we revised both the Greedy Heuristic and enhanced Depth First Search to include the consideration of connectivity and obstacle awareness constraints.

5.4.1 Greedy Heuristic

As shown in Algorithm 3, we propose a Greedy Heuristic approach to cover the maximum number of grid points within our monitored space \mathbf{A} . A while loop is implemented to check if any cover-ble 3D regions and candidate placement locations are available (line 1).

In the *for* loop we evaluate each candidate location C_L in \mathbf{L} , checking to see if it is within the communication range, R_{max} (line 3). This is implemented to ensure the connectivity withing the network and follows our constraint requirement. For each *face*, we check each monitoring point which is within the covered field by current candidate location C_L to see whether it is obstructed by any obstacle in \mathbf{O} or not using the Accurate Detection method as seen in Figure 5.4. We compute all the *face* directions for the candidate locations C_L and pick the *face* direction which covers the maximum number of monitoring points F (line 4-5). After the *for* loop, we record the C_L which covers the maximum number of monitoring points in \mathbf{L} . Instead of arbitrarily choosing a location in \mathbf{L} to deploy a video sensor, in each iteration of the *while* loop, the greedy heuristic algorithm strives to choose among the candidate locations in \mathbf{L} and find the location where the deployed video sensor can cover the maximum number of lattice points F (lines 11). After the candidate location is chosen by the greedy heuristic algorithm and a video sensor is deployed at that location to cover a number of grid points that have not been covered by any video sensor, the remaining points that need to be covered by a C_L is recalculated iteratively. The next candidate location C_L to be selected will cover the maximum number of coverable grid points. The final list of $S = \{(L_{p_1}, D_{p_1}), (L_{p_2}, D_{p_2}), \dots (L_{p_n}, D_{p_n})\}$ that will fully cover the 3D monitored space is then returned (line 14).

5.4.2 Enhanced Depth First Search

Similar to our initial approach in the deployment problem, we attempt to reduce the search space in \mathbf{L} , (i.e., lattice grid) as seen in Algorithm 4, by using branch pruning to cut the candidate location C_L options that will not produce feasible branches that can improve the solution. Let *depth* be defined as the search branches that require equal or more number of video sensors to cover the space than *best* i.e., the *final* list of (C_L, D) sets from our greedy heuristic algorithm which is the first enhancement. We determine its feasibility by comparing the current search branch to our *best* found solution, (line 2). We next check to

Algorithm 3 Greedy Heuristic Algorithm for Connectivity and Obstacle Awareness

Input: \mathbf{A} , \mathbf{D} and \mathbf{L} **Output:** Set of S with max F in descending order*Initialize* : List of A , D , L and *final*.

```
1: while  $\mathbf{L} > 0$  and  $\mathbf{A} > 0$  do
2:   for each  $C_L$  in  $\mathbf{L}$  do
3:     if  $C_L \in$  connected final video list then
4:       Compute all  $face \in D$  for  $C_L$  considering  $\mathbf{O}$ 
5:       Select  $face$  with max  $F$ .
6:       if  $C_L(F)^* > C_L(F)$  then
7:         Update  $C_L$ ;
8:       end if
9:     end if
10:  end for
11:  record  $C_L$  with max  $F$  into final
12:  remove  $C_L$  from  $\mathbf{L}$  and update  $\mathbf{A}$ 
13: end while
14: return final
```

see if there are remaining points to be covered in \mathbf{A} , if no we return the optimal and exit the program; otherwise we continue (line 5-7).

To further improve the solution our second enhancement selects the location in each iteration by sorting all of C_L in descending order based on the maximum number of grid points that the video sensor placed at the specific deployment position can cover and then produces the facing direction $face$ for the set (C_L, D) with consideration to the obstacles (lines 9-14). The advantage of this strategy is we can reduce the infeasible solution options to quickly find better quality candidate location options and avoid overhead cost of memory and other resources. A similar enhancement approach is used when selecting the facing direction of a video sensor. We ensure for Algorithm 4, that all the candidate location in the list have a communication path to the base-station L_0 (line 18). A tight lower bound enhancement is provided to approximate the number of video sensors required with relative certainty that we need to fully cover all remaining grid points with S . The new estimates for the number of required sensors S is recorded and stored for processing in the algorithm. An update is done where we remove the candidate position in \mathbf{L} where video sensors have already

Algorithm 4 Enhanced-DFS Algorithm for Connectivity and Obstacle Awareness

Input: \mathbf{A} , \mathbf{L} and \mathbf{D}

Output: Minimized set of $|S| = n$, where S is optimal

Initialize : List of \mathbf{A} , \mathbf{D} , \mathbf{L} and optimal.

```
1: DFS(depth)
2: if depth  $\geq$  best then
3:   return
4: end if
5: if  $A == 0$  then
6:   Update best;
7:   Record optimal;
8: else
9:   for each  $C_L$  in  $\mathbf{L}$  do
10:    Compute all  $face \in D$  for  $C_L$  considering  $\mathbf{O}$ 
11:    Select  $face$  with max  $F$ .
12:    Store  $C_L$  with max  $F$ .
13:   end for
14:   Sort  $\mathbf{L}$  by descending order of  $F$ ;
15:   Store  $C_{L^*} \rightarrow Queue$ ;
16:   while  $Queue \neq \emptyset$  do
17:      $C_{L^*} \leftarrow Dequeue$ ;
18:     if  $C_L \in$  connected final camera list then
19:       if ( $lowerBound(C_L(F)^*) + depth \geq optimal$ ) then
20:         break;
21:       end if
22:       Record  $C_L^*$  and  $F^*$ 
23:       Remove  $C_L^*$  from  $\mathbf{L}$  and  $F^*$  from  $\mathbf{A}$ ;
24:       DFS(depth + 1)
25:       Add  $F^*$  back to  $A$ ;
26:     end if
27:   end while
28:   Add all removed  $C_L^*$  back to  $\mathbf{L}$ 
29: end if
30: return optimal
```

been deployed and remove the covered points from \mathbf{A} (lines 19-23). Next, we recursively call the enhanced DFS search to explore the remaining search branches (line 24). Our proof is revised to include the area of interest within \mathbf{A} considering both the connectivity and obstacle constraint. The lower bound for the enhanced DFS algorithm is defined by the following constraint:

$$\sum_{i=1}^n (C_L, D)_{F_c} \geq 0, \text{ and } n + depth \geq best \quad (5.4)$$

where n is incremented by one until this constraint is satisfied. A minimized set of $|S| = n$, where S is optimal is returned as shown in line 30.

Theorem 2. *Given enough time, the enhanced DFS algorithm can return the optimal solution for the discrete version problem.*

Proof. We generalize the traditional DFS algorithm as a brute force approach that searches each branch until an optimal solution is returned. We assume our discrete version problem to be a brute force approach with some of the infeasible solution space reduced. Thus, we will prove that our pruning feature in the enhanced DFS algorithm will not eliminate optimal solutions. We select and expand on search branches by determining the maximum number of fresh points covered by the chosen location and direction. Given that $(C_L, D)_{F_c}$ is non-negative where $F_c \geq 0$ and if current $n + depth \geq best$, we do not need to check it further. As a result, we can cut the branch. Specifically, since the number of F is recalculated after each iteration of the search, the cost function is non-decreasing as the search step traverses the graph for a feasible solution. \square

5.5 Experimental Results

We conduct extensive simulations to evaluate our solutions using the customized simulator implemented by Java Script. Table 5.2 outlines the default parameter settings that are utilized in the performance evaluation. For comparison, we extend the random

Table: 5.2 Simulation Default Parameters II

Parameter	Denotation	Default Value
3D Indoor Space	L	Length X 60 X 100
Monitored Area	A	Same as 3D Indoor Space
Obstacle	\mathbf{O}	$20 \times 20 \times 30$
Deployment Area	Tuple(L,D)	Top half of walls and ceiling
Granularity of \mathbf{A}	g_A	20
Granularity of \mathbf{L}	g_L	25
Granularity of \mathbf{D}	g_D	45°
Field of View	FOV	50°
Max Sensing Range	R_S	100
Max Communication Range	R_{max}	100
Aspect Ratio	*	1.778
Near Field	*	1
Far Field	*	100

Table 5.2. Connectivity and Obstacle Awareness: Simulation Parameters

algorithm used in Chapter 4 to support connectivity and obstacle awareness, which is used as the baseline approach. The baseline algorithm will select an initial location, then the facing direction of the video sensor is adjusted to greedily cover the maximum number of fresh grid points (i.e. those points that have not been covered by previously deployed video sensors). If the video sensor cannot cover any fresh points, it will be removed and another random location will be selected. This process will continue until all the grid points in \mathbf{A} have been covered. An example of the testing environment is simulated in Figure 5.17, where an optimal deployment scenario is shown. For each setting, we run the baseline algorithm 100 times and show the average of the results with an error bar to indicate the minimum and maximum values.

We first examine how our solutions perform with different lengths of the 3D indoor space. The results are shown in Figure 5.6. As expected, with the dimension of the 3D indoor space increasing, more wireless video sensors are required to cover the indoor space. However, our greedy heuristic algorithm produced a 38% reduction in the number of video sensors required to cover the monitoring area. It is worth noting that although we run

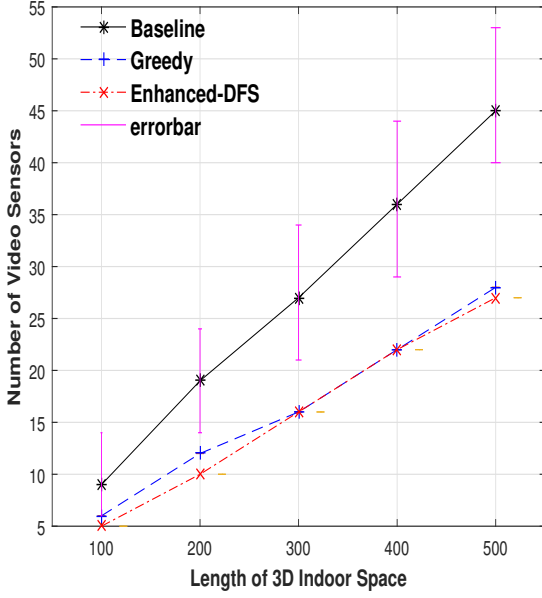


Figure 5.6. Variations of Indoor Space

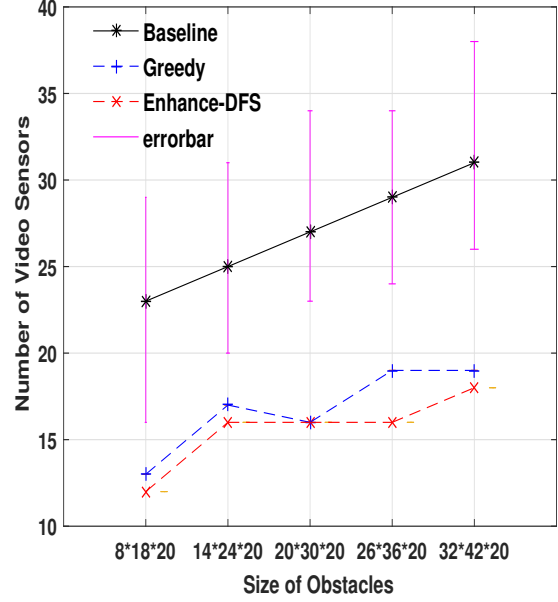


Figure 5.7. Varying Obstacle Dimensions

the simulation on a standard PC with configurations listed as: i7 processor, 16GB RAM, 1TB SSD and set a short time limitation (1 *hour*) for the enhanced DFS solution in our evaluation, it can still successfully return the optimal results for the cases with the length of 3D indoor space equal to 100 and 200. We thus conjecture that our enhanced DFS solution can return optimal results for more cases if it is run on a local or cloud server that has much more computation power and more time is allowed. We next investigate how obstacles can affect the performance of our solutions, which is illustrated in Figure 5.7 and Figure 5.8.

It is easy to see that in Figure 5.7, as the dimension of obstacles increases, the line-of-sight of a video sensor is more easily blocked, requiring more video sensors needed to fully cover the 3D indoor monitoring space. On the other hand, Figure 5.8 shows that more distance between neighboring obstacles indicates less chance to block the line-of-sight of video sensors (i.e, in terms of length, for example, every 50 unit of length has one obstacle) making the number of required video sensors become smaller. Nevertheless, in both figures, the number of required video sensors by our greedy and enhanced DFS solutions is much smaller than that of the baseline algorithm. In particular, the reduction over the baseline is

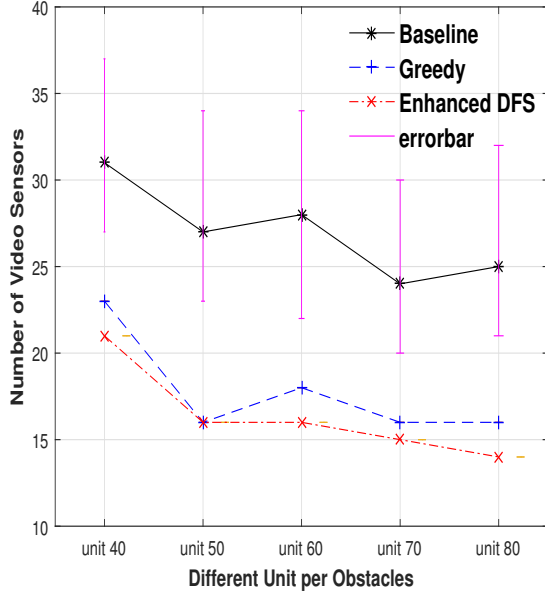


Figure 5.8. Obstacle per Unit Length

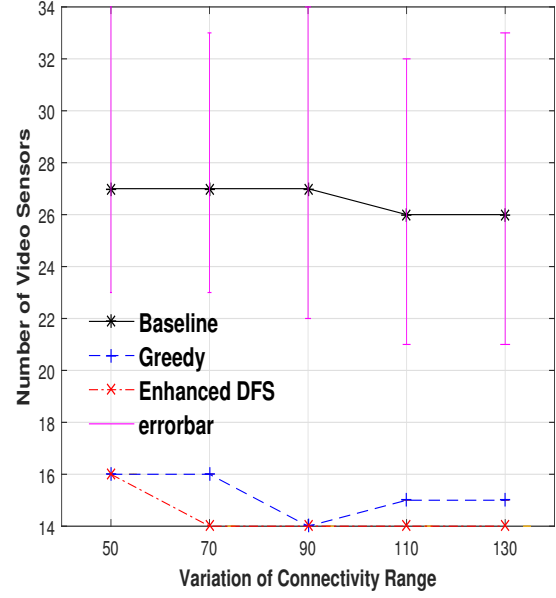


Figure 5.9. Varying Connectivity Range

up to 44% for the greedy solution and the enhanced DFS solution can further reduce up to another 18%.

Figure 5.9 shows the impact of the maximum communication range on the deployment of video sensors. When the maximum communication range is smaller, more video sensors are required to maintain the network connectivity, which can make for densely deployed sensors in terms of coverage. On the other hand, when the maximum communication range is larger, its impact to the deployment becomes limited, where the coverage affects more. As illustrated in both Figure 5.10 and Figure 5.11, a dependency relationship is evident in both cases. It is easy to see that the number of sensors increase with the rising of the near field and decreases with a larger *FOV* setting. Figure 5.12 shows the reduction of video sensors required to cover the space as the distance among neighboring grid increases, this occurs because it takes less grid points to fill the space as the distance grows between them. Figure 5.13 shows the stable performance of both the enhanced DFS and greedy heuristic, where improvements in reducing S over the greedy is achieved.

Finally, in the testing environment that evaluated the domain variance of the candi-

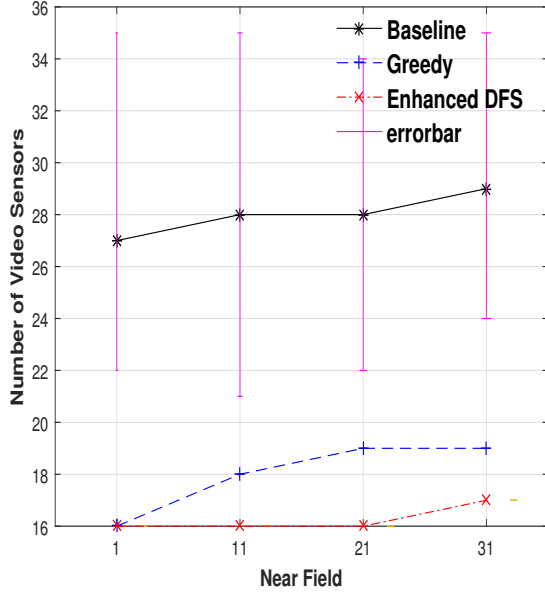


Figure 5.10. Impact of Near Field

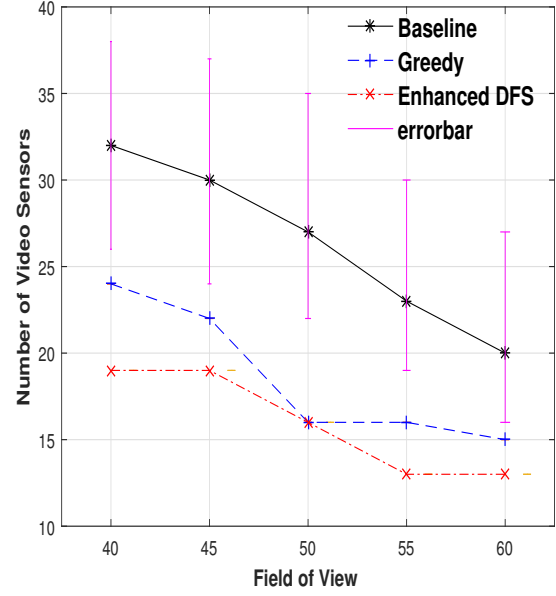


Figure 5.11. Variation of Field of View

date location as seen in Figure 5.14 and granularity of L in Figure 5.15, an optimal solution for the enhanced DFS algorithm was returned before the 1 hour time limit restriction for the simulation ended. Moreover, the number of required video sensors for both our greedy and enhanced DFS solutions is much less than that of the baseline solution, where our greedy solution can reduce the required video sensors by up to 50% over the baseline algorithm as seen in Figure 5.14, and our enhanced DFS solution can further achieve an additional reduction on the number of required video sensors by up to 21% over the greedy solution as shown in Figure 5.11. Figure 5.16 shows the obstacle awareness aspect of the model in action, where only points within the sensing range are covered and the points within the shadow of the obstacles are uncovered. Additionally, Figure 5.17 presents an aerial perspective of a completed deployment.

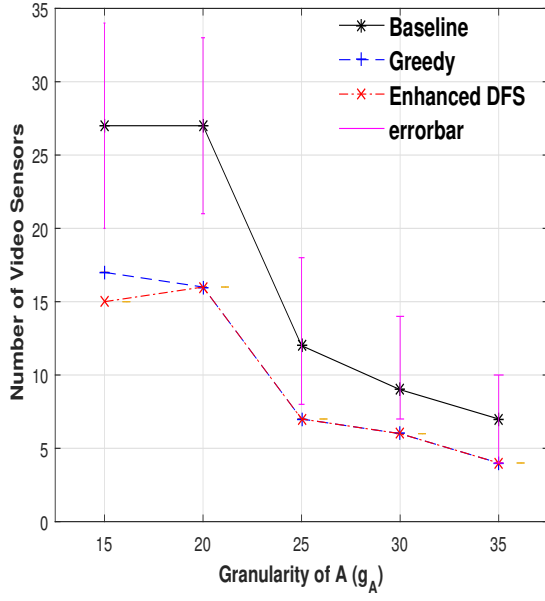


Figure 5.12. Varying g_A

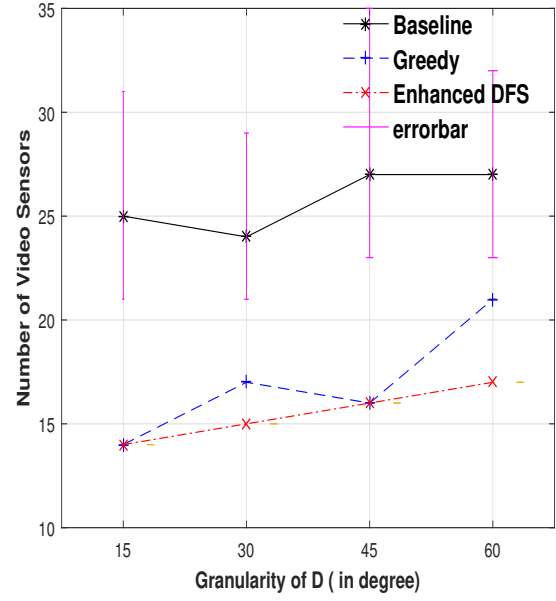


Figure 5.13. Varying g_D

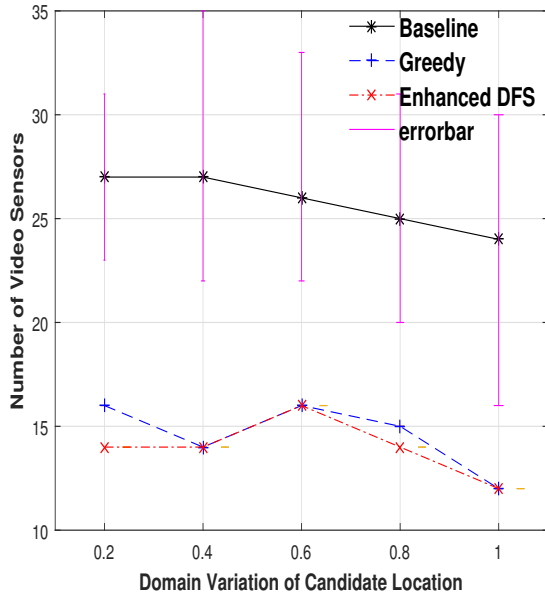


Figure 5.14. Varying Impact of C_L

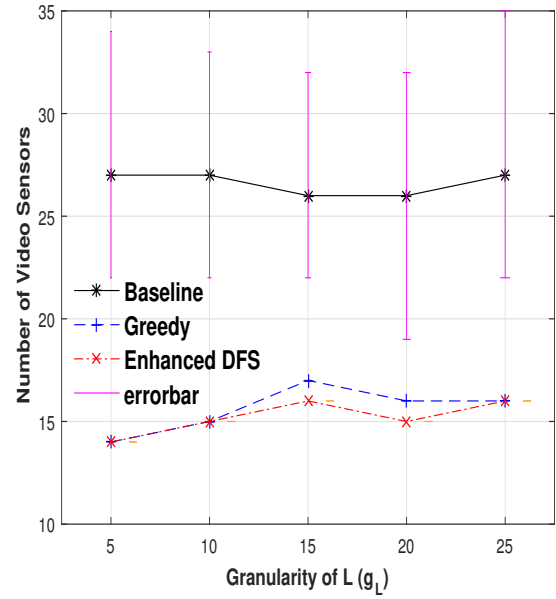


Figure 5.15. Varying Impact of g_L

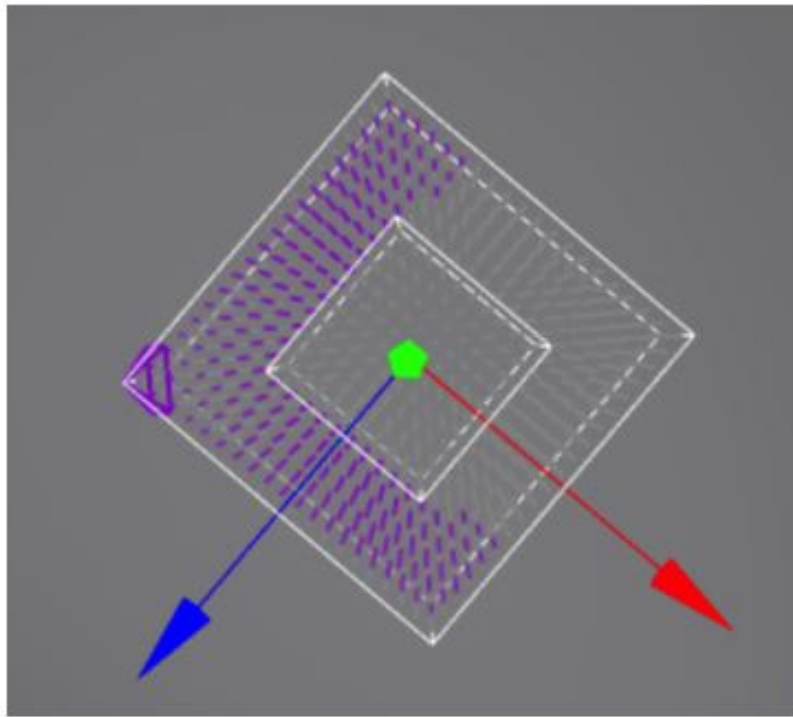


Figure 5.16. Simulated Deployment Scenario considering Single Obstacle

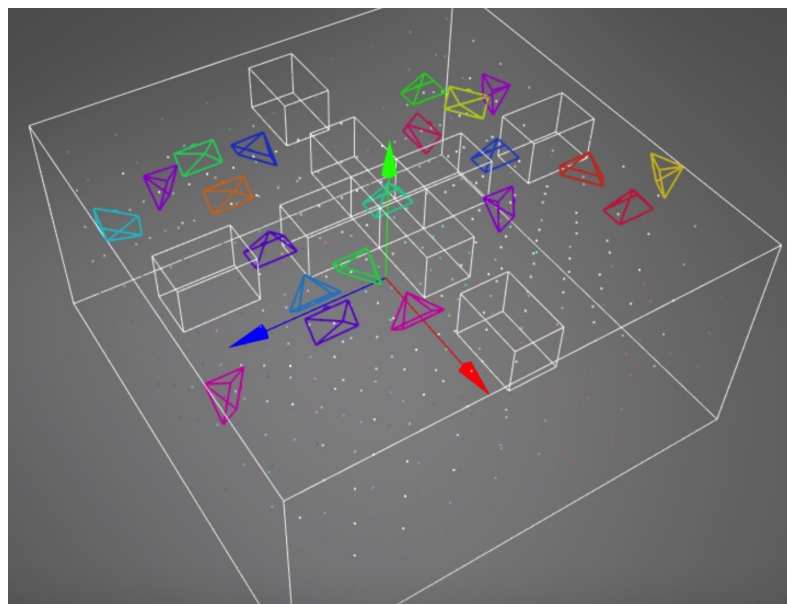


Figure 5.17. Simulated Deployment Scenario considering Multiple Obstacles

CHAPTER 6

2-ANGULAR-COVERAGE

In this chapter, we discuss the coverage problem for WVSNs. The coverage problem can be formulated as an optimization problem, thus different optimization techniques can be employed to solve it. Additionally, the generalized “coverage problem” term can be interpreted in multiple ways and in different contexts (i.e., full, target and k-coverage). Herein, we compare the aforementioned categories of different coverage approaches and the motivation of our proposed angular coverage problem in Section 6.2. Section 6.3.1 and Section 6.3.2 will introduce our network models extended from Chapter 5 to consider the additional coverage constraint imposed in the continuous and discrete space for 2-angular coverage. A detailed discussion of the solutions is presented in Section 6.4, where we identify our distinct methodology from existing approaches. The performance evaluation of both the greedy heuristic and enhanced depth first search algorithms are presented in Section 6.5.

6.1 Introduction

Wireless Video Sensor Networks are prominently featured in countless technological applications including: image processing, site monitoring and intruder detection systems. These video sensor nodes are established to work autonomously and have the ability to communicate directly over a shared wireless channel. Given these features, the use of wireless video sensor networks in many industrial sectors have grown exponentially [133]. Furthermore, wireless video sensor networks have a niche within this field because of the uniqueness attributed to its directional sensing range [105]. Such factors makes the integration of this technology quite an interesting research topic. One of the fundamental research issues for WVSNs is the coverage problem in wireless video sensor networks. Coverage in wireless

video sensor networks is an indicator used to measure how well and for how long the sensors are able to monitor the physical space. Though, extensive research has been conducted in the area of WVSNs to provide techniques that will optimize sensor coverage, existing literature studies heavily focus on only three types of coverage schemes, namely area (i.e., a full coverage model), target (i.e., a point coverage models) and k-coverage (i.e., a degree of coverage model). A detailed review of each is presented in the next section.

6.2 Coverage Comparison

In this section, we discuss some of the commonly utilized WVSN coverage models and identify problematic factors that can arise when directly applying strategies already well developed for the general coverage problem (i.e., traditional disk sensing models do not consider the intrinsic directional property of video sensors), to a scenario where consideration to angular coverage is required. We discuss the potential design conflicts that are problematic for each scheme in depth.

Area coverage in WVSNs is an approach utilized, where video sensor nodes are deployed in such a way that the area “covered” by the network is maximized [54]. In area coverage models, the objective is essentially define as “Given the sensing range R of sensors, how to place the sensors so that the entire monitored area will be fully covered?” as illustrated in Figure 6.1b, while the deployment problem is to address how to place sensors so that the the number of sensors N needed to cover the monitoring area is minimized. [76]. In this scenario, an issue can arise when multiple sensors are deployed to cover a monitored space where it is assumed that all sensors will have the same omnidirectional sensing ranges. This can produce overlapping of covered spaces (i.e., redundant spaces) occurring due to densely deployed sensor within the monitored region as seen in [134; 135]. This is a contributing factor to network in-efficiency and excessive resource allocation where the trade-off is a fully covered monitored space but with high resource overhead. Consequently, this approach implements a scheme that can only provide area coverage of the monitored space but

does not fully ensure fault tolerance and optimally within the WVSN. Several of the studies discussed above utilize omnidirectional 2D network models. However, the work in [136] studies the full-coverage problem for three dimensional sensor networks. Their approach designed a set of patterns to achieve area coverage in addition to one other sensor network metric. The goal of the work was to prove the optimality of the deployment strategy among all of the regular lattice deployment patterns presented (i.e., right parallelepiped, basic and body centered lattices). Though, the deployment strategy considered the three dimensional aspect of the deployment region, the work did not employ a realistic approach to the sensing range as it utilized an omnidirectional model.

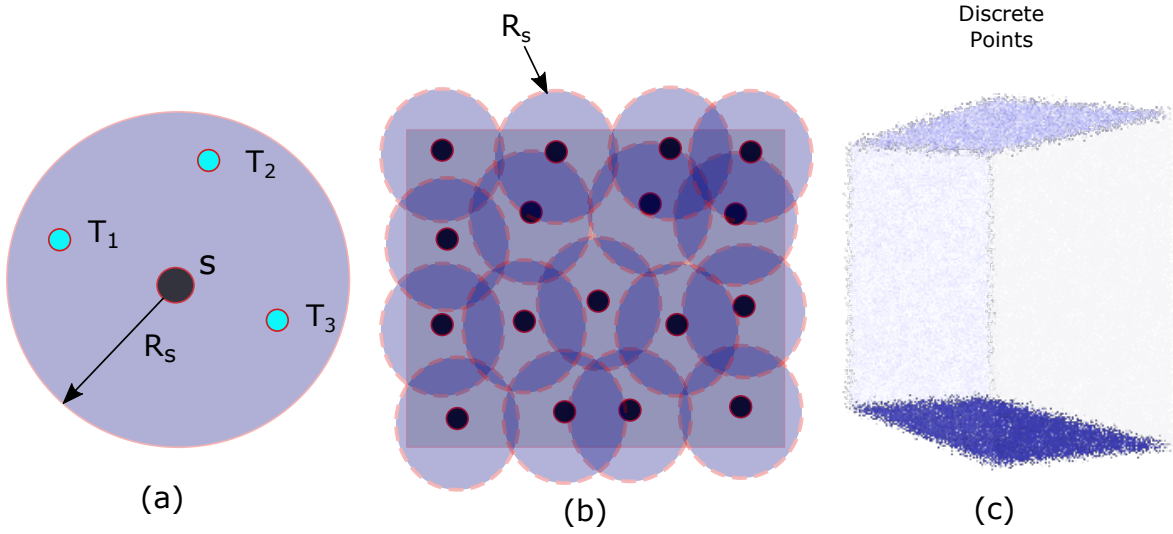


Figure 6.1. Coverage Model Comparison (a) Target Coverage (b) Area Coverage (c) Barrier Coverage using pointillism of discrete points [4]

The objective of the *target coverage* problem is to employ a technique to monitor a target set T such that all the targets within T will be covered, if it is covered by at least one sensor node [137]. In target coverage, targets are often modeled as a set of discrete space points within the sensing field as shown in Figure 6.1a. The points are then used to represent

a physical target or static event [4].

Traditional schemes for this problem often utilize a disk sensing method. Within this method, it is assumed that a target within the sensing disk range is always detected within a probability of one, however, if the target (i.e., points) is outside the disk, it is not detected. The implementation of this approach strategy is based on an unrealistic sensing model that assumes near perfect coverage within the circular disc. However, in real-world applications, the sensing properties of sensors are greatly affected by environmental factors (i.e., walls and other obstacles) and physical phenomena (i.e., distance and interference). Thus, there is a need for more accurate coverage models which can characterize the properties attributed to more practical deployment environments such as those found within buildings and other personal spaces. Furthermore, the limited orientation angles of target sensing models can render coverage effective only from certain view points producing restricted viewpoints, occlusion of other elements in the environment including other targets and an inadequate coverage model for larger targets [138]. Hence, we take into consideration the angular aspect of the video sensor and consider both the connectivity and obstacles as it relates to 3D indoor settings.

Over the years, there have been several studies that have proposed solutions to specifically address the *k-coverage* problem, such as works in [139] and [140]. Though, not as extensively explored in comparison to target and area coverage, existing literature highlights the importance of evaluating the problem. The *k-coverage* metric allows for a more tailored, practical and concise approach to monitor and measure the performance of the network environment, whereby a region is minimally *k*-covered by a set of sensors S , such that each point in the given region is “monitored” by at least k distinct sensors. Figure 6.1c presents *k*-covered discrete points using the pointillism method (i.e, covering in which small, distinct points in various colors are applied in patterns) where different colors are assigned to point coverings within the network, identifying the specific (i.e., distinct) sensor that covers it. The nature of this proposed design promotes fault-tolerance within the network as presented in

[141]. However, majority of the existing literature employ simple 2D models, fails to support fault tolerance while promoting network efficiency (i.e., strives to deploy distinct sensors for k -coverage). As a result, their applications may not be practical in real world settings and thus incur inconsistencies.

As stated previously, k -coverage is a unique coverage metric, capable of demonstrating the coverage quality measured in terms of coverage ratios or a probability guarantee. However, other factors weigh heavily on the performance of the WVSN. Such metrics are often overlooked when analyzing the quality of coverage within wireless video sensors networks. A different metric is needed to measure the quality of the deployment in terms of coverage.

Video sensor nodes are established to work autonomously and have the ability to communicate directly over a shared wireless channel in WVSNs [142]. Given these features, the use of wireless video sensor networks has expanded into many fields. Furthermore, the Internet of Things (IoT) is an emerging research area centered on smart physical devices which communicate, coordinate and collect data utilizing the Internet [133]. Wireless video sensor networks have a niche within this field because of the uniqueness attributed to its directional sensing range [105]. Potential areas of extension includes providing an outlet to incorporate control of networks vital to business, home and intruder monitoring [143]. Coupled with the economical cost of video sensors, this preliminary area within the IoT will allow for immediate integration into systems for personal surveillance via mobile and online applications. This is our motivation for the the work. In particular, the development of new trends which facilitate the monitoring of indoor spaces utilizing the Internet of Things (IoT) paradigm coupled with the integration of WVSNs is needed to advance the field of video surveillance applications. Consequently, this approach requires a scheme that will provide not only optimal coverage of the monitored area but additionally ensures fault tolerance within the system.

Due in part to the uniqueness of the video sensor directional sensing range, we propose

to exploit the directional feature to determine the optimal angular-coverage of each deployed video sensor. In this chapter, we study this challenging problem by considering 2-coverage for WVSNs in 3D indoor space monitoring building on the work from Chapter 4 (i.e., optimal deployment) and Chapter 5 (i.e., connectivity and obstacle awareness). Moreover, we propose to deploy the video sensors from divergent directional angles and further extend 2-coverage to “2-angular-coverage” to imply such unique requirements in WVSNs. To solve this problem, we propose bilateral solutions namely a greedy heuristic and an enhanced Depth First Search approach (E-DFS). Initially, by formulating the problem into a continuous space and then formulating the problem into a discrete model where we partition the surveillance region into grid subareas. Additionally, we develop a strategy to determine the degree of coverage for the WVSN, where a given location needs to be covered by at least 2 video sensors and fulfill the angular-coverage requirement. In the sections below, we present the problem formulation and both the continuous space and discrete space models for the 2-angular coverage problem.

6.3 Problem Formulation

The monitoring area is assumed to be a three dimensional indoor space (e.g., a departmental floor of a building). Also, within the indoor environment, there are often restrictions for prime deployable locations (e.g., ceilings, wall space not obstructed by furniture placement) which introduce additional challenges. Our goal is thus to optimize the placement and facing direction of each video sensor so as to minimize the number of video sensors required to fully monitor all the required 3D areas. Below, we address the WVSN coverage problem for 3D indoor space with consideration of 2-angular coverage for video sensor deployment.

6.3.1 Continuous Space Model

As highlighted previously, introducing WVSNs into an environment presents additional challenges that are not often attributed to WSNs such as coverage quality and the resolution of the image which depends on the orientation of the video sensor. Our work, differs from the existing literature by considering the unique angular coverage feature and

real world 3D modeling for WVSNs indoor monitoring. Moreover, we create a scheme to determine the minimum number of video sensors to achieve 2-angular coverage (i.e., adapted from 2-coverage) from different directional perspectives. Presented below is our continuous space model extend to consider 2-angular coverage.

To recall, we let \mathbf{A} denote the 3D areas that must be continuously monitored (i.e., covered) by the video sensor, \mathbf{S} . Now, let \mathbf{L} denote the deployable positions within the monitored area that can be used to deploy (i.e., place video sensor at the specified location) the video sensors. The location and direction of the video sensor is defined as a tuple (L, D) . We assume for a 3D indoor space, the location L is represented as a three dimensional coordinate for the video sensor placement (x, y, z) and the direction $D = (x', y', z')$ is a point on the surface of a unit sphere (which throughout the paper will be referred to as the facing direction sphere) with its radius equal to 1 and centered at $(0, 0, 0)$. We denote *face* to represent the facing direction vector from $(0, 0, 0)$ to (x', y', z') . As mentioned earlier, the interest areas specifically within the 3D area in \mathbf{A} that we want to monitor is denoted as a . Additionally, we assume R_S is the maximum sensing range of the video sensor and R_{max} denotes the maximum communication range of the video sensor. We model the view frustum of the video sensor with parameters: *farfield*, *nearfield*, *field of view* (FOV) and *aspect ratio*. Thus, we formulate our problem as to find a set of locations and directions of video sensor nodes where, $S = \{(L_1, D_1), (L_2, D_2), \dots, (L_n, D_n)\}$, subject to the following additional constraints:

(1) Sensor Location Constraint:

$$\forall (L, D) \in \mathbf{S}, L \in \mathbf{L} ;$$

(2) Area Coverage Constraint:

$$\begin{aligned} \forall a \in \mathbf{A}, \exists (L, D) \in S, \text{ such that,} \\ a \in \text{Cover}(L, D, R_S) \text{ and} \\ \forall O \in \mathbf{O}, \text{Segment}(La) \cap O = \emptyset ; \end{aligned}$$

(3) Network Connectivity Constraint:

$$\begin{aligned} \exists \text{Path} = \{(L_{p_1}, D_{p_1}), (L_{p_2}, D_{p_2}), \dots, (L_{p_k}, D_{p_k})\}, \\ \text{such that } \text{Path} \subseteq S, L_{p_1} = L, \\ |L_{p_i} L_{p_{i+1}}| \leq R_{max} \text{ for } i = 1 \dots (|\text{Path}| - 1) ; \\ |L_{p_k} L_0| \leq R_{max} ; \end{aligned}$$

(4) Region Angular 2-Coverage Constraint:

$$\begin{aligned} \forall a \in \mathbf{A}; \\ \exists (L_i, D_i), (L_j, D_j) \in \mathbf{S} \ (i \neq j) ; \text{ such that,} \\ a \in \mathbf{C}(L_i, D_i, R_S) \cap \mathbf{C}(L_j, D_j, R_S) \text{ and} \\ \angle(\overrightarrow{aL_1 \rightarrow L_i}, \overrightarrow{aL_2 \rightarrow L_j}) \geq \theta \end{aligned}$$

, where θ is a predefined angle ¹. The regional angular 2-coverage constraint specifies that the monitoring point a is in the covered frustum of video sensors deployed at both L_i and L_j to allow for 2-angular coverage. The concept model for the 2-angular coverage constraint is illustrated in Figure 6.2. This constraint will allow for fault tolerance within the network, where if one sensor fails another sensor will be available to continue to monitor the space. Moreover, when two video sensors work simultaneously, the 2-angular coverage can greatly

¹For this work, the angle θ default value was equivalent to 120°.

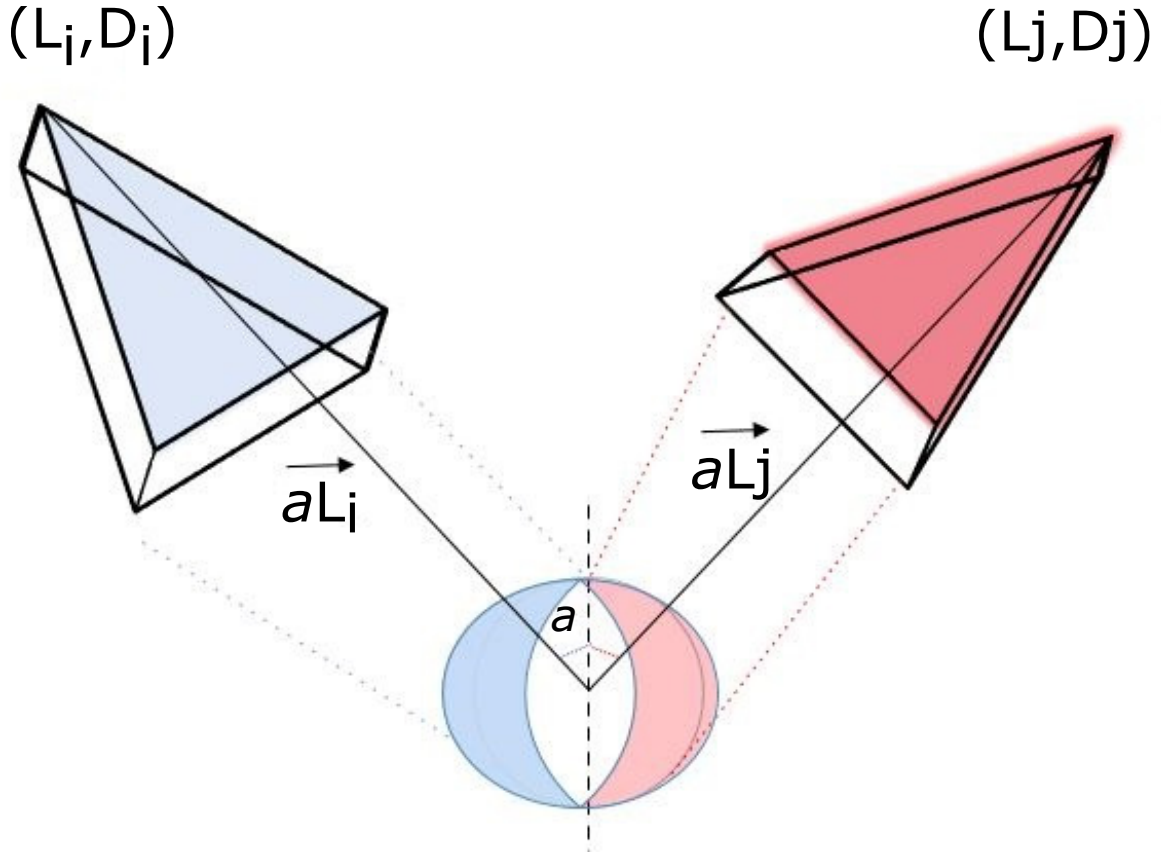


Figure 6.2. Concept Model for 2-angular-coverage

enrich the captured video information if an object presents at a . Our objective is thus to minimize $|S| = n$, where n is the number of video sensors deployed to cover the monitored area.

As formulated in the angular 2-coverage constraint, there must exist at least 2 video sensors that can monitor a given point to provide varied angular coverage. This feature will allow for potential applications in tracking of a target and facial recognition analysis (i.e., facial tracking) by providing image captures from a 2-angular perspective. Consider this scenario where the requirement is to provide some type of facial tracking in a given monitored space. The monitored space can be defined into regions within a video sensor network, where the camera placement should provide different angles of the tracked target. Though overlapped visible objects may occur with the WWSN target tracking, the image

can provide completely different scenes considering the angle. Based on this assumption we propose to consider if the video sensor captures the point from different viewing directions for which the angle between their direction is at least greater than or equal to 120° a predefined difference in the coverage angle. A representation of 2-coverage with the video sensor direction constraint for a point in the monitored area is shown in Figure 6.2 vectors $\overrightarrow{aL_i}$ and $\overrightarrow{aL_j}$.

6.3.2 Discrete Space Model

A revised network lattice model is used to approximate the continuous space model by dividing the monitored space \mathbf{A} that must be fully covered into discrete 3D grids. We require that all grid points in the region are at least 2 – *angular* covered. This technique is also used for the deployable positions in \mathbf{L} to divide each area that can be used to deploy the video sensors. Let L_C denote the candidate locations where a video sensor can be deployed in the 3D monitored space. The tuple (L_C, D) is used to denote the candidate location and direction of a video sensor. The distance between two neighboring grid points is denoted as g_A , which is used to quantify the granularity of the grids used in the monitored space \mathbf{A} and g_L denotes the granularity of the grids used in \mathbf{L} . In addition, we also divide the surface of the facing direction sphere into grids (like the longitudes and latitudes divide the surface of the earth) and use g_D to denote the granularity. We assume that a wireless video sensor can only face a direction where its D falls on a grid point, which represents the deployment of a sensors with consideration given to the facing direction sphere techniques. It is easy to see that by adjusting the three granularity parameters, we can easily achieve the required accuracy for approximating the continuous space model.

6.4 Experimental Methodology

In this section, we tackle the 2-angular coverage problem for 3D indoor monitoring by extending the greedy heuristic and enhanced Depth First Search algorithm proposed in the previous chapters.

6.4.1 Greedy Heuristic Algorithm

The greedy algorithm works in an iterative manner to deploy video sensors as seen in Algorithm 5. In each iteration, the algorithm strives to choose among the available locations in L and find the location where the deployed video sensor can cover a maximum (i.e., greedy approach) number of fresh points. Next, we then update whether the points that are covered can be categorized as: 1-coverage, 2-angular-coverage or no coverage. This is implemented by recalculating the point coverage for each video sensor at a deployable location in the monitored area. By continuously tracking and updating the points covered, we determine the minimum number of remaining location in L to place the deployable video sensor. Additionally, we then consider those points that are only covered by one video sensor and check if by placing a video sensor at a location, which deployment achieves the maximum conversion of points from 1-coverage to 2-coverage by an opposing video sensor and via an angle different from the previous coverage (as required by the 2-angular-coverage)². The one that can maximally cover the most points considering the 2-angular coverage constraint after recalculation will then be chosen as the next location to deploy a video sensor.

6.4.2 Enhanced Depth First Algorithm

In Algorithm 6, we provide our enhanced DFS algorithm to further improve the quality of our solution. In a traditional depth first search approach, the exploration of each branch within our solution set will need to be searched to pick a location in \mathbf{L} and a facing direction D . However, this is not ideal since the solution space can quickly expand with the size of \mathbf{L} . In this solution, a pruning technique is implemented which can exploit the search by cutting off and removing most of the solution space that does not include high-quality solutions. We are able to efficiently reduce the size of the candidate location L_C options by starting the search from the result of our greedy heuristic algorithm as the currently found *best* solution. The search branches (*depth*) that already have used equal or more number of

² We additionally consider the obstacle constraint in line 4 of Algorithm 5, however we emphasize angular coverage in this chapter. The connectivity constraint is upheld in line 3.

Algorithm 5 Greedy Heuristic Algorithm for 2-Angular Coverage

Input: \mathbf{A} , \mathbf{D} and \mathbf{L} **Output:** Set of S with max F in descending order*Initialize* : List of \mathbf{A} , \mathbf{D} , \mathbf{L} and *list*.

```
1: while  $\mathbf{L} > 0$  and  $\mathbf{A} > 0$  do
2:   for each  $L_C$  in  $\mathbf{L}$  do
3:     if  $L_C \in$  connected video sensor list then
4:       Compute all  $face \in D$  for  $L_C$  considering 2-angular coverage
5:       Select  $face$  with max  $F$ .
6:       if  $L_C(F)^* > L_C(F)$  then
7:         Update  $L_C$ ;
8:       end if
9:     end if
10:  end for
11:  record  $L_C$  with max  $F$ 
12:  remove  $L_C$  from  $\mathbf{L}$  and update  $\mathbf{A}$ 
13: end while
14: return list
```

video sensors compared to the currently found best solution can be pruned (lines 1-4).

When a branch is searched incrementally where it is either fully explored or pruned, the search will revert back to its previous status to be popped from the stack. Additionally, the selection of the location in each iteration is chosen by using the default order (e.g., sort all L_C by decreasing order) based on the maximum number of points that a video sensor at a deployable location can convert a point from non-coverage to 1-coverage and from 1-coverage to 2-angular-coverage and then choose the first element in the sorted list (lines 8-14). This approach allows our algorithm to quickly find high quality solutions and skip as many low quality solutions as possible. When a search branch is pruned or explored fully, the search will update all the parameters and will continue the search using its previous status (lines 18-22). The search will halt after all points in the 3D indoor space is 2-angular-covered. An optimal set of video sensor deployment locations is then returned (line 26).

The lower bound for the enhanced DFS algorithm is defined as:

$$lowerbound = \min(S_1, \dots, S_n) \quad (6.1)$$

Algorithm 6 Enhanced-DFS Algorithm for 2-Angular Coverage

Input: \mathbf{A} , \mathbf{L} and \mathbf{D}

Output: Minimized set of $|S| = n$, ensuring 2-angular-coverage

Initialize : list of \mathbf{A} , \mathbf{D} , \mathbf{L} and *list*.

```
1: DFS(depth)
2: if depth  $\geq$  best then
3:   return
4: end if
5: if  $A == 0$  then
6:   Update best;
7: else
8:   for each  $L_C$  in  $\mathbf{L}$  do
9:     Compute face for  $L_C$ .
10:    Select face with 2-coverage.
11:    Store  $L_C$  selected.
12:   end for
13:   Sort  $\mathbf{L}$  by descending order;
14:   Store  $L_C \rightarrow \text{Queue}$ ;
15:   while  $\text{Queue} \neq \emptyset$  do
16:      $L_C \leftarrow \text{Dequeue}$ ;
17:     if  $L_C \in \text{list}$  then
18:       if ( $\text{lowerBound}(L_C) + \text{depth} \geq \text{list}$ ) then
19:         break;
20:       end if
21:       DFS(depth + 1)
22:       Add removed undeployed  $L_C$  back to  $\mathbf{L}$ 
23:     end if
24:   end while
25: end if
26: return list
```

, where n is incremented by one until $f(x) \leq \text{min}$. We then use the minimum number of the video sensors that we have chosen during this process as the lower bound, where $f(x)$ is calculated:

$$f(\text{min}) = \sum_{S \in \mathbf{S}}^n (x)^S F \quad (6.2)$$

In each search step, if this lower bound plus the number of video sensors that we have already deployed is equal to or greater than the currently found best (i.e., In the greedy

algorithm, we calculate a number of cameras for the covering. We use this number as our baseline for the enhanced DFS algorithm. At the first iteration, the “best” term equals the baseline) solution, the search branch can be safely prune (line 2).

We utilize the following theorem.

Theorem 3. *The enhanced-Depth First Search (e-DFS) proposed in Algorithm 6 returns the optimal solution for the 2-angular coverage for the discrete version of the general problem.*

Proof. To prove the statement, we assume that if the search branches are not pruned, the solution will continue to search exhaustively and return the optimal solution. The proof is thus proved by showing that the branch pruning in-fact does not miss the optimal deployment placement, since the number of the locations that we have chosen during this process F is the lower bound, those paths will contain the most current search branch. Additionally, if the lower bound plus the number of video sensors that we have already deployed is greater than or equal to (i.e. \geq) the currently found best solution, the search branch can be safely pruned.

□

6.5 Experimental Results

The efficiency and validity of our algorithms are evaluated by extensive simulations. The simulation settings are provided in Table 6.1. We use a similar simulation environment to the previous chapters. However, the random baseline algorithm used in both of the previous chapters failed to return a solution in a feasible time frame for this study. We hypothesize that the random deployment approach may have lead to an uneven distribution of video sensor nodes within the monitored region.

Due to this factor, coupled by the rigorous additional constraints implemented in Chapter 5 where the video sensor nodes deployed must have a path to the sink and not be obstructed by an obstacle resulted in a greatly reduced solution space. The solution space (i.e., available deployable sensors compliant with all constraints) was further reduced when

Table: 6.1 Simulation Default Parameters III

Parameter	Denotation	Default Value
3D Indoor Space	L	Length X 60 X 100
Monitored Area	A	Same as 3D Indoor Space
Obstacle	\mathbf{O}	Size of $O \in \mathbf{O} = 20 \times 20 \times 30$
Deployment Area	Tuple(L,D)	Top half of walls and ceiling
Coverage Angle	$2 - angular - coverage$	$\geq 120^\circ$
Granularity of \mathbf{A}	g_A	20
Granularity of \mathbf{L}	g_L	25
Granularity of \mathbf{D}	g_D	45°
Field of View	FOV	50°
Max Sensing Range	R_S	100
Max Communication Range	R_{max}	100
Aspect Ratio	*	1.778
Near Field	*	1
Far Field	*	100

Table 6.1. 2-Angular Coverage: Simulation Parameters

the constraint of the sensor pairings for opposing angles of greater than or equal to 120° was added. Thus, the initial uneven distribution produced a scenario where none or a limited number of sensors were in enough proximity of each other to fit into the 2-angular coverage constraint. After fulfilling the prior three constraints from our previous models, no feasible solution could be produced using the baseline algorithm.

We now present a discussion on the performance evaluation of our algorithms. Figure 6.3 shows how the number of video sensors required to fully cover a space decrease as the domain variations of the 3D indoor space deployable regions increase (i.e., the allowable regions for deployment on the walls and ceilings are increased by a factor of .20), which highlights the flexibility of the system. The greedy heuristic and enhanced-DFS algorithms produce similar results initially, then the enhanced-DFS algorithm outperforms the greedy heuristic algorithm, specifically when the domain of the indoor space reaches the .80 interval. In Figure 6.4, we evaluated how varying the field of view for the video sensor impacts the performance of both algorithms. As expected, the required number of video sensor to

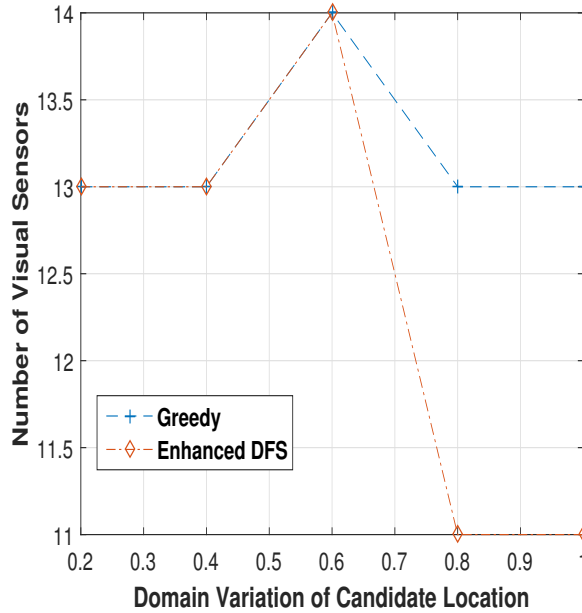


Figure 6.3. Variation of Deployment Domain

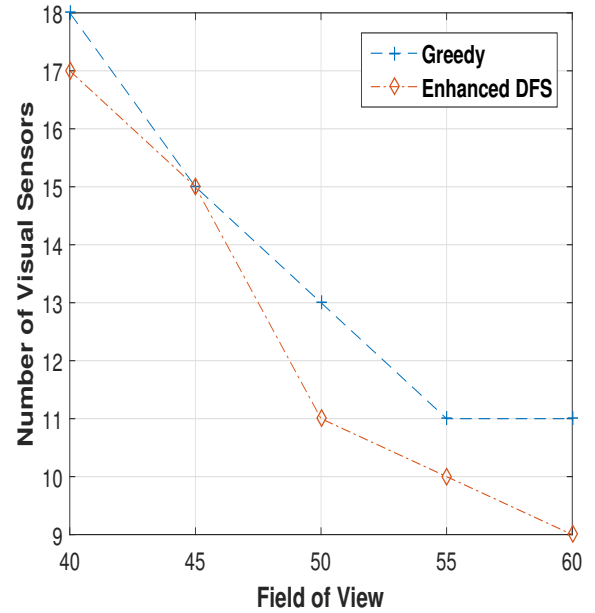


Figure 6.4. Varying Field of View

cover the space steadily decreases as the field of view angle is increased (i.e, angle of view expanded), this is seen in both the greedy heuristic and enhanced DFS, in this instance the enhanced DFS out performs the greedy approach.

We evaluate how varying the near field impacts our algorithmic solutions in Figure 6.5. Initially, both algorithms require the same amount of required video sensors to cover the 3D indoor space. This changes as the near field parameter is increased. We can see that both algorithms require an increase amount of sensors to fully cover the 3D area as the near field distance is increased. The increase is intuitively expected because of the near field parameter properties. The near field corresponds to the distance to the near clipping plane of the video sensor's viewing frustum. As the near field distance increases, the eye of the video sensor moves further away from the viewing frustum near plane. This results in an angular and volume reduction for the sensor viewing frustum which results in a decrease in sensing ability thus requiring more sensors to monitor the area. This is shown in the figure by the unstable (i.e., non -linear) results. However, both algorithms were able to generate feasible results.

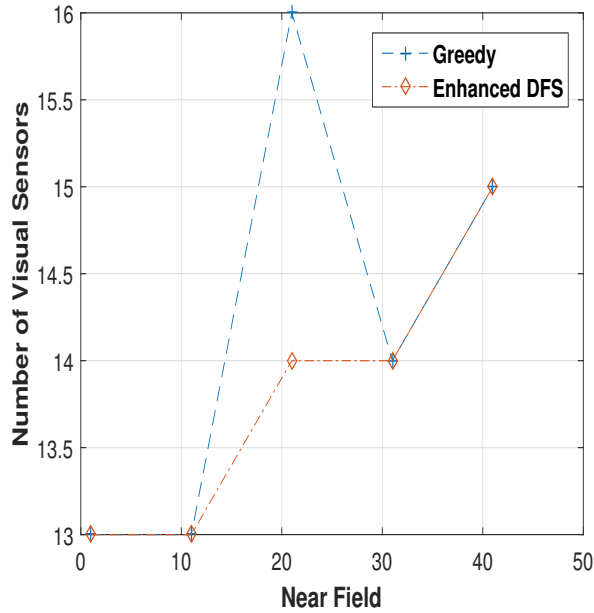


Figure 6.5. Varying Near Field

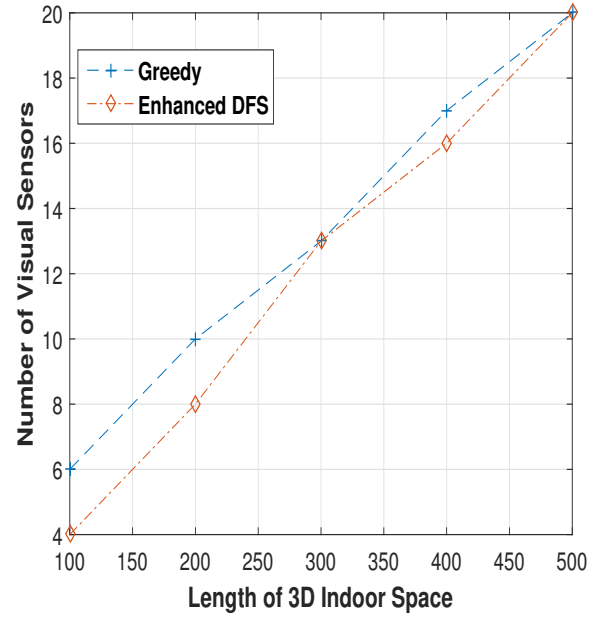


Figure 6.6. Varying 3D Indoor Space

Figure 6.6 presents an evaluation on how varying the 3D indoor space sizes impact the performance of both the greedy heuristic and enhanced depth first search algorithms. As the monitoring space increases the number of video sensors required to cover the space is relatively similar. Additionally, as the size of the required space grows, so does the number of sensors (i.e., the number of sensors required to cover a larger space is expected). Overall, the enhanced DFS algorithm requires less video sensors in comparison to the greedy solution.

Granularity is the extent to which data is broken down into smaller parts. For the evaluation of the granularity of deployable location points in \mathbf{L} within the discrete grid model, we evaluated how g_L impacts the greedy heuristic and enhanced depth first search algorithms. The enhanced DFS algorithm appears to out perform the greedy heuristic algorithm in the beginning portion of performance evaluation as seen in Figure 6.7, however, as the granularity of the locations for candidate deployments L_C increases, so does the number of video sensors that are required to fully cover the 3D indoor space, this is true for both algorithms where both have similar results. We know that due to the increase in possible deployable locations, the solution space is now larger. In this case, a scenario may happen where our implemented

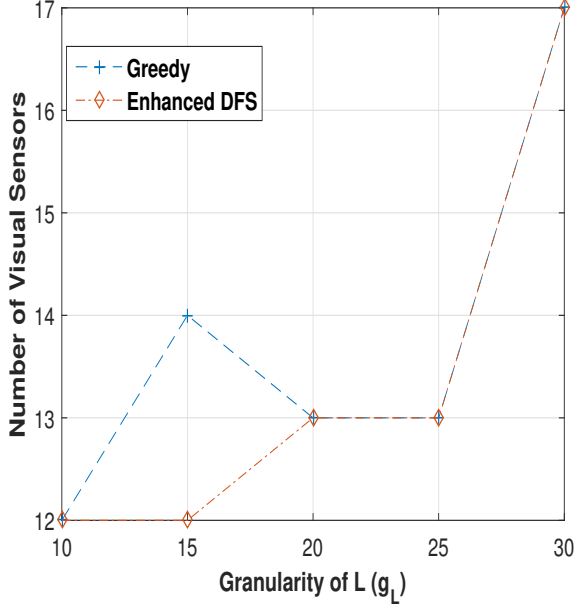


Figure 6.7. Impact of Varying g_L

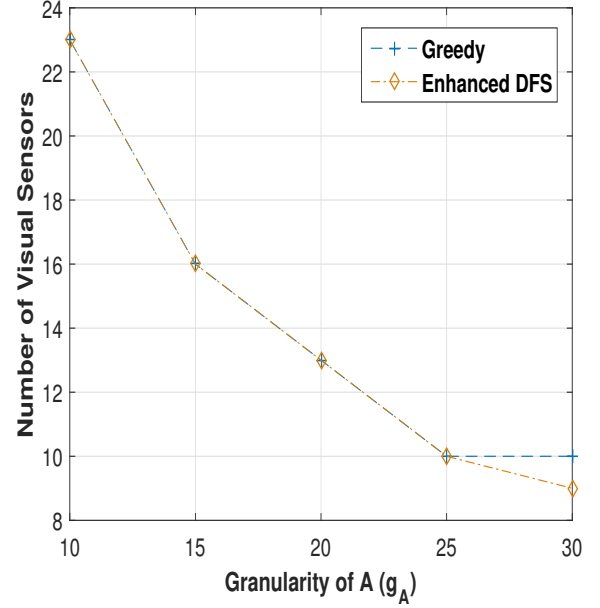


Figure 6.8. Varying Impact of g_A

method for both algorithms continuously checks to see if a new deployable location can better (i.e., cover more new fresh points) cover the deployment region considering the constraints. We thus expected the number of sensors required to fully cover the area would decrease due to the increase in number of deployable locations (i.e., more options for sensor deployment). In Figure 6.8, we look at how the granularity of \mathbf{A} affects the performance of our algorithms. The results show that as the granularity of \mathbf{A} increases, the number of video sensors required to cover the region decreases. Both algorithms perform relatively similar but at the ending parameter interval of 30, the enhanced DFS performs better. In this case, initially in the evaluation the grid points to be covered in \mathbf{A} are sparsely placed in the environment requiring more sensors to fully cover the area. However, once the grid points are densely distributed (i.e., an increase in g_A) through out the monitored space, a fewer number of sensors are required to fully cover the area.

We now look at how varying the far field impacts the performance of the algorithmic solutions as presented in Figure 6.9. Both algorithms have comparable performances as it relates to the number of video sensors required to monitor the coverage area. The figure shows

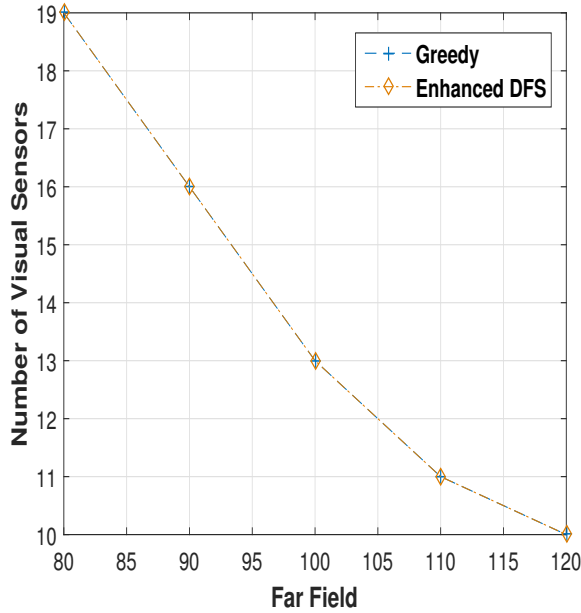


Figure 6.9. Impact on Varying the Far Field

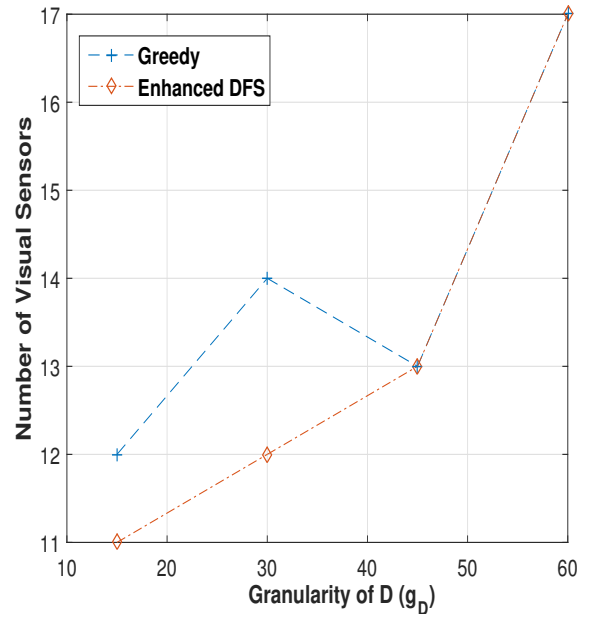


Figure 6.10. Impact of Varying g_D

that as the far field parameter increases, the number of required sensors in fact decreases. This is opposite of what we experienced in the performance evaluation for the near field parameter. This is due to the reversal of projected view as it corresponds to sensor view frustum. When the far field parameter is increased, the volume and viewing angle of the view frustum increases (i.e., far clipping plane expands to a larger viewing capacity), too. This allows for increase sensing and the ability to cover a larger area within the video sensor's view frustum. Thus, reducing the number of sensors required to cover the 3D space. Next, we evaluated how varying g_D would impact our solutions. The granularity of D refers to the facing direction sphere (i.e., granularity of the longitude and latitude lines on the facing sphere) for the video sensor. In Figure 6.10, the greedy heuristic algorithm requires more sensors to cover the space while considering 2-angular coverage.

Additional testing was conducted to evaluate both algorithms. Overall, the greedy algorithm results were analyzed and the number of required video sensors was comparable to our enhanced DFS algorithm. However, in several instances the enhanced DFS out performed the greedy heuristic algorithm and in one scenario the enhanced DFS was able to reduce the

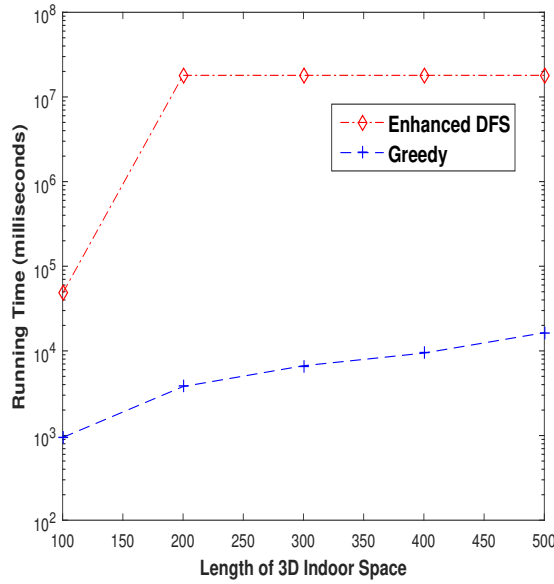


Figure 6.11. Running Time

number of video sensors by up to 22% over our greedy algorithm.

In Figure 6.11, we further evaluated the running time performance of the greedy heuristic and enhanced depth first search algorithms. The greedy heuristic algorithm was able to successfully complete with running times of .948, 3.722, 6.676, 9.424 and 16.398 (i.e., in seconds for the case lengths of 100, 200, 300, 400 and 500, respectively) with relatively stable performance. We hypothesize that the algorithm was able to return in the allotted time, given its selection of locally optimal solutions. As seen in Figure 6.11, the enhanced DFS algorithm returned an optimal solution in case interval 100 with a running time of 48.058 seconds. However, during the run time evaluation cases of 200, 300, 400 and 500, respectively, the algorithm did not finish within the set 30 minute time limit. We hypothesize that due to the gradual increase in the length of space for monitoring cases 200-500, our enhanced depth first algorithm was not able to finish within the allotted time frame. Though the algorithm did not return the optimal solution for the aforementioned cases above, the enhanced DFS algorithm was able to improve the solution of the greedy heuristic algorithm most of the time for a practical real world monitoring setting as shown in Figure 6.6. Overall, the

greedy heuristic algorithm was able to return a solution more efficiently with the trade-off of optimality. In reference to the running time of the enhanced DFS, the rapid growth of n (i.e., input size) increased the search space exponentially in some cases resulting in a longer running time.

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1 Conclusion

Wireless Video Sensor Networks have real world implications in many important fields. This is seen in the increased development of practical applications for areas such as campus, industrial, medical and environmental monitoring, which greatly benefit from the technology. Many existing studies offer optimization techniques to improve the functionality of the network as it relates to deployment. However, most existing literature on the topic often model the deployment environment with two dimensional aspects and assign omnidirectional sensing ranges to the sensor, without giving consideration to the 3D aspects of true deployment environments for real world applications.

This dissertation, studied the deployment problem for 3D indoor space monitoring in WVSNs with the considerations to connectivity, obstacle awareness and 2-angular coverage. For the basic deployment problem discussed in Chapter 4, we presented techniques to optimize the placement and angular direction of each video sensor in a continuous space, and then converted the model into a discrete version where a grid model was presented to replicate a more precise real world indoor environment, allowing a fine grained approach to achieve arbitrary approximation precision in the models. The greedy heuristic algorithm we proposed computed locally optimal solutions and our enhanced DFS algorithm was further proposed to improve the coverage of all regions in the monitored space whereby it minimized the number of video sensors and can used branch pruning to efficiently return optimal solutions for the locational position and directional angle of the video sensors.

Based on the high quality solutions yielded from the deployment problem perfor-

mance evaluation, the discrete space model was extended to include constraints to consider jointly the connectivity and obstacle metrics for WVSNs, discussed in Chapter 5. To ensure connectivity within the deployed video sensor network a *communication path* is established between a start node to the base station, which is sufficient enough to supply connectivity with WVSN. Two strategies are proposed to account for obstacles within the monitored environment. The Divide and Conquer scheme utilized a straight line equation which resulted in some obstacles not being detected. So, the Accurate Detection strategy was developed to provide improved obstacle awareness within the monitored environment. This technique used a more precise model to fully detect all obstacles within the monitored environment. The Accurate Detection model was then implemented in both the greedy heuristic and enhanced depth first search algorithms. Extensive performance evaluations were then conducted using the custom simulation environment. The enhanced DFS algorithm significantly reduced the number of video sensors required to fully cover the 3D indoor monitoring space, returning optimal solutions, given a reasonable amount of time (i.e., 30 minutes).

We studied the 2-angular-coverage problem for WVSNs in Chapter 6, where we developed a scheme to effectively monitor a 3D indoor space from different perspectives using visual sensor networks. For each perspective, we ensured that all the points inside the interest areas were covered. In each candidate location for a visual sensor, we discretize the area to effectively deploy 2-angular-coverage visual sensors. By iteratively selecting the facing direction choices for each visual sensor our enhanced-DFS algorithm further improved the coverage of the space and provided feasible solutions for networks beyond the greedy heuristic algorithm.

7.2 Future Work

In an attempt to develop future enhancements to our formulated solutions, alternative techniques can be utilized for the deployment optimization methods. One example of an alternative technique is the development of an optimization algorithm based on the “intelli-

gent” foraging behaviour of different swarm insects (i.e., such as the honey bee) [144] [145] [146] [147]. The Artificial Bee Colony algorithm is applied to many optimization problems by converting the problem into an objective function which determines the best parameter vector. Then, the artificial bee solution randomly discovers a population of initial solution vectors and then iterative improves upon them by employing various strategies that select improved solution spaces by searching neighboring locations while abandoning poor solutions. It is thus very interesting to further investigate how such optimization techniques can be applied to the problems in our scenario to improve the performance.

Another future research trend is to consider the traffic dynamics that may be commonly seen in WVSNs. Traffic modeling in WVSNs is dependent upon the type of data patterns that are presented to the particular application within the network. Past research on the topic suggests that the traffic patterns are categorized as event driven or periodic burst throughout the network. In the case of the periodic bursts of data, the use of constant bit rate (CBR) is used to model the arrival of data. However, if the data within the network arrives at a variable time frame, a Poisson process can be used to model the data traffic. As for the scenario of event driven data, a more tailored approach is needed. Some of the more common strategies include: Pareto Distribution Process, Weibull Distribution Process, Regression and Transform-expand-sample (TES) models [148]. In the Pareto approach the inter-arrival times of data are calculated independently and identically distributed (IID). The Weibull Distribution Process uses an ON/OFF model to monitor the burst patterns in the source data traffic. The TES model can capture the correlated sequences and marginal distribution of data within the network. Within a Regression Model, predictors and response parameters are introduced where the relationship among the variables are analyzed and characterization of the outcome (prediction) is developed. An extension to this work may consist of incorporating in-network data traffic models to provide more practical traffic routing and scheduling solutions.

Additionally, in Chapter 5, we discussed the use of different models for the connectiv-

ity metric as it relates to WVSNs. During the discussion, we explored both the deterministic and probabilistic models. Ultimately, we used a deterministic model for the network connectivity constraints. To extend the model, a potential extension would be to use a probabilistic communication model for our problem, which can make the model more practical. The aforementioned topics above and their application in the improvement of WVSNs is the subject of future work.

7.3 Research Publications

The following publications resulted from the work studied in this dissertation.

1. Brown, T., Wang, Z., Shan, T., Wang, F. and Xue, J., 2016, March. On wireless video sensor network deployment for 3D indoor space coverage. In SoutheastCon, 2016 (pp. 1-8). IEEE [149]
2. Brown, T., Wang, Z., Shan, T., Wang, F., and Xue, J. 2017. Obstacle and Connectivity Aware Wireless Video Sensor Deployment for 3D Indoor Monitoring: Poster Abstract. In Proceedings of the Second International Conference on Internet-of-Things Design and Implementation (IoTDI '17). ACM, New York, NY, USA, 305-306. DOI: <https://doi.org/10.1145/3054977.3057924> [150]
3. Wang, Z., T. Brown, T. Shan, F. Wang, and J. Xue. "On 2-Angular-Coverage in Wireless Visual Sensor Network Deployment for 3D Indoor Monitoring." In Proceedings of the SouthEast Conference, pp. 145-148. ACM, 2017. [151]
4. Brown, T., Wang, Z., Shan, T., Wang, F., and Xue, J., Obstacle Aware Wireless Visual Sensor Network Deployment for 3D Indoor Space Monitoring, Submitted for Review to IEEE Globecom '17.
5. Brown, T., Wang, Z., Shan, T., Wang, F., and Xue, J. Obstacle Aware Analysis of k-Coverage, Connectivity and Traffic in Wireless Visual Sensor Network Deployment

for 3D Indoor Space Monitoring, To Be Submitted Springer PPNA Special Issue on
Network Coverage: From Theory to Practice (SI: Network Coverage) (Journal)'18.

BIBLIOGRAPHY

BIBLIOGRAPHY

- [1] F. Heart, A. McKenzie, J. McQuillan, and D. Walden. (1978) Arpanet completion report, bolt, beranek and newman. [Online]. Available: <https://web.archive.org/web/20120306185838/http://som.csudh.edu/cis/lpress/history/arpamaps/>
- [2] S. L. Inc. (2013) The Evolution of wireless sensor networks. [Online]. Available: <http://www.silabs.com/Support%20Documents/TechnicalDocs/evolution-of-wireless-sensor-networks.pdf>
- [3] F. Heart, A. McKenzie, J. McQuillan, and D. Walden. (2013) New 3g sensors stream photo and video to the cloud for new security and military applications. [Online]. Available: <http://www.libelium.com/video-camera-3g-sensors-stream-photo-video-to-cloud-for-security-military-applications/>
- [4] A. Shan, X. Xu, and Z. Cheng, “Target coverage in wireless sensor networks with probabilistic sensors,” *Sensors*, vol. 16, no. 9, p. 1372, 2016.
- [5] F. Aurenhammer, “Voronoi diagrams—a survey of a fundamental geometric data structure,” *ACM Computing Surveys (CSUR)*, vol. 23, no. 3, pp. 345–405, 1991.
- [6] A. Gogu, D. Nace, A. Dilo, and N. Mertnia, “Optimization problems in wireless sensor networks,” in *Complex, Intelligent and Software Intensive Systems (CISIS), 2011 International Conference on*. IEEE, 2011, pp. 302–309.
- [7] C. Buratti, A. Conti, D. Dardari, and R. Verdone, “An overview on wireless sensor networks technology and evolution,” *Sensors*, vol. 9, no. 9, pp. 6869–6896, 2009.
- [8] F. Nack, “An overview on wireless sensor networks,” *University, Barlin*, 2014.
- [9] Q. Wang and I. Balasingham, *Wireless Sensor Networks - An Introduction*. INTECH Open Access Publisher, 2010.
- [10] C. Zhu, L. Shu, T. Hara, L. Wang, S. Nishio, and L. T. Yang, “A survey on communication and data management issues in mobile sensor networks,” *Wireless Communications and Mobile Computing*, vol. 14, no. 1, pp. 19–36, 2014.
- [11] X. Dong, M. C. Vuran, and S. Irmak, “Autonomous precision agriculture through integration of wireless underground sensor networks with center pivot irrigation systems,” *Ad Hoc Networks*, vol. 11, no. 7, pp. 1975–1987, 2013.

- [12] L. Liu, R. Wang, D. Guo, and X. Fan, "Message dissemination for throughput optimization in storage-limited opportunistic underwater sensor networks," in *Sensing, Communication, and Networking (SECON), 2016 13th Annual IEEE International Conference on*. IEEE, 2016, pp. 1–9.
- [13] M. C. Vuran and I. F. Akyildiz, "Cross-layer packet size optimization for wireless terrestrial, underwater, and underground sensor networks," in *INFOCOM 2008. The 27th Conference on Computer Communications*. IEEE. IEEE, 2008, pp. 226–230.
- [14] I. T. Almalkawi, M. Guerrero Zapata, J. N. Al-Karaki, and J. Morillo-Pozo, "Wireless multimedia sensor networks: current trends and future directions," *Sensors*, vol. 10, no. 7, pp. 6662–6717, 2010.
- [15] S. Mini, S. K. Udgate, and S. L. Sabat, "Sensor deployment and scheduling for target coverage problem in wireless sensor networks," *IEEE Sensors Journal*, vol. 14, no. 3, pp. 636–644, 2014.
- [16] B. Shahrokhzadeh, M. Dehghan, and M. Shahrokhzadeh, "Improving energy-efficient target coverage in visual sensor networks," *Journal of Computer and Robotics*, vol. 10, no. 1, pp. 53–65, 2017.
- [17] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, "A survey on wireless multimedia sensor networks," *Computer networks*, vol. 51, no. 4, pp. 921–960, 2007.
- [18] S. Xu, W. Lyu, and H. Li, "Optimizing coverage of 3d wireless multimedia sensor networks by means of deploying redundant sensors," *International Journal of Advanced Studies in Computers, Science and Engineering*, vol. 4, no. 9, p. 28, 2015.
- [19] D. S. Deif and Y. Gadallah, "Classification of wireless sensor networks deployment techniques," *IEEE Communications Surveys Tutorials*, vol. 16, no. 2, pp. 834–855, Second 2014.
- [20] K. Han, L. Xiang, J. Luo, and Y. Liu, "Minimum-energy connected coverage in wireless sensor networks with omni-directional and directional features," in *Proceedings of the thirteenth ACM international symposium on Mobile Ad Hoc Networking and Computing*. ACM, 2012, pp. 85–94.
- [21] A. Raha, S. Maity, M. K. Naskar, O. Alfandi, and D. Hogrefe, "An optimal sensor deployment scheme to ensure multi level coverage and connectivity in wireless sensor networks," in *Wireless Communications and Mobile Computing Conference (IWCMC), 2012 8th International*.
- [22] C.-F. Huang, Y.-C. Tseng, and L.-C. Lo, "The coverage problem in three-dimensional wireless sensor networks," *Journal of Interconnection Networks*, vol. 8, no. 03, pp. 209–227, 2007.
- [23] T. Andersen and S. Tirthapura, "Wireless sensor deployment for 3d coverage with constraints," in *Networked Sensing Systems (INSS), 2009 Sixth International Conference on*. IEEE, 2009, pp. 1–4.

- [24] K. Y. Chow, K. S. Lui, and E. Y. Lam, "Achieving 360° angle coverage with minimum transmission cost in visual sensor networks," in *2007 IEEE Wireless Communications and Networking Conference*, March 2007, pp. 4112–4116.
- [25] E. Horster and R. Lienhart, "Approximating optimal visual sensor placement," in *2006 IEEE International Conference on Multimedia and Expo*, July 2006, pp. 1257–1260.
- [26] Y. Morsly, N. Aouf, M. S. Djouadi, and M. Richardson, "Particle swarm optimization inspired probability algorithm for optimal camera network placement," *IEEE Sensors Journal*, vol. 12, no. 5, pp. 1402–1412, 2012.
- [27] X. Zhang, X. Chen, J. L. Alarcon-Herrera, and Y. Fang, "3-d model-based multi-camera deployment: A recursive convex optimization approach," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 6, pp. 3157–3169, Dec 2015.
- [28] A. Newell, K. Akkaya, and E. Yildiz, "Providing multi-perspective event coverage in wireless multimedia sensor networks," in *Local Computer Networks (LCN), 2010 IEEE 35th Conference on*, Oct 2010.
- [29] N. Khalil, M. R. Abid, D. Benhaddou, and M. Gerndt, "Wireless sensors networks for internet of things," in *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2014 IEEE Ninth International Conference on*. IEEE, 2014, pp. 1–6.
- [30] P. J. Denning, "The science of computing: The arpanet after twenty years," *American Scientist*, vol. 77, no. 6, pp. 530–534, 1989.
- [31] A. Fraboulet, G. Chelius, and E. Fleury, "Worldsens: development and prototyping tools for application specific wireless sensors networks," in *Information Processing in Sensor Networks, 2007. IPSN 2007. 6th International Symposium on*. IEEE, 2007, pp. 176–185.
- [32] I. F. Akyildiz, W.-Y. Lee, and K. R. Chowdhury, "Crahns: Cognitive radio ad hoc networks," *AD hoc networks*, vol. 7, no. 5, pp. 810–836, 2009.
- [33] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer networks*, vol. 52, no. 12, pp. 2292–2330, 2008.
- [34] B. Liu, O. Dousse, P. Nain, and D. Towsley, "Dynamic coverage of mobile sensor networks," *IEEE Transactions on Parallel and Distributed systems*, vol. 24, no. 2, pp. 301–311, 2013.
- [35] S. Kisseleff, I. F. Akyildiz, and W. H. Gerstacker, "Throughput of the magnetic induction based wireless underground sensor networks: Key optimization techniques," *IEEE Transactions on Communications*, vol. 62, no. 12, pp. 4426–4439, 2014.
- [36] Y. Noh, U. Lee, P. Wang, B. S. C. Choi, and M. Gerla, "Vapr: void-aware pressure routing for underwater sensor networks," *IEEE Transactions on Mobile Computing*, vol. 12, no. 5, pp. 895–908, 2013.

- [37] F. Nadeem, E. Leitgeb, M. S. Awan, and S. Chessa, “Comparing the life time of terrestrial wireless sensor networks by employing hybrid fso/rf and only rf access networks,” in *2009 Fifth International Conference on Wireless and Mobile Communications*, Aug 2009, pp. 134–139.
- [38] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, “Wireless multimedia sensor networks: Applications and testbeds,” *Proceedings of the IEEE*, vol. 96, no. 10, pp. 1588–1605, 2008.
- [39] S. Misra, M. Reisslein, and G. Xue, “A survey of multimedia streaming in wireless sensor networks,” *IEEE communications surveys & tutorials*, vol. 10, no. 4, 2008.
- [40] A. R. Al-Ali, Y. R. Aji, H. F. Othman, and F. T. Fakhreddin, “Wireless smart sensors networks overview,” in *Second IFIP International Conference on Wireless and Optical Communications Networks, 2005. WOCN 2005.*, March 2005, pp. 536–540.
- [41] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A survey on sensor networks,” *IEEE Communications magazine*, vol. 40, no. 8, pp. 102–114, 2002.
- [42] S. Soro and W. B. Heinzelman, “On the coverage problem in video-based wireless sensor networks,” in *Broadband Networks, 2005. BroadNets 2005. 2nd International Conference*.
- [43] A. Seema and M. Reisslein, “Towards efficient wireless video sensor networks: A survey of existing node architectures and proposal for a flexi-wvsnp design,” *IEEE Communications Surveys & Tutorials*, vol. 13, no. 3, pp. 462–486.
- [44] J. Kaur. Comparison of traditional versus video sensor nodes. [Online]. Available: <http://www.slideshare.net/jass91/data-aggregation-in-wireless-sensor-networks>
- [45] H. Cho, Y. Baek, and C. M. Kyung, “Wireless video sensor network platform and its application for public safety,” in *2014 IEEE Intl Conf on High Performance Computing and Communications, 2014 IEEE 6th Intl Symp on Cyberspace Safety and Security, 2014 IEEE 11th Intl Conf on Embedded Software and Syst (HPCC,CSS,ICSS)*, Aug 2014, pp. 743–746.
- [46] X. Liu, “Coverage with connectivity in wireless sensor networks,” in *Broadband Communications, Networks and Systems, BROADNETS 2006. 3rd International Conference*, pp. 1–8.
- [47] W. Y. Poe and J. B. Schmitt, “Node deployment in large wireless sensor networks: Coverage, energy consumption, and worst-case delay,” in *Asian Internet Engineering Conference*, ser. AINTEC ’09. New York, NY, USA: ACM, 2009, pp. 77–84.
- [48] M. Cardei and J. Wu, “Coverage in wireless sensor networks,” *Handbook of Sensor Networks*, vol. 21, 2004.
- [49] F. Bauer, “Normalized graph laplacians for directed graphs,” *Linear Algebra and its Applications*, vol. 436, no. 11, pp. 4193–4222, 2012.

- [50] S. J. M. Baygi, M. Mokhtari *et al.*, “Evaluation performance of protocols leach, 802.15.4 and cbrp, using analysis of qos in wsns,” *Wireless Sensor Network*, vol. 6, no. 10, p. 221, 2014.
- [51] D. G. Costa and L. A. Guedes, “The coverage problem in video-based wireless sensor networks: A survey,” *Sensors*, vol. 10, no. 9, pp. 8215–8247, 2010.
- [52] P. Sahu and S. R. Gupta, “Deployment techniques in wireless sensor networks,” *International Journal of Soft Computing and Engineering*, vol. 2, no. 3, pp. 525–526, 2012.
- [53] O. Khatib, “Real-time obstacle avoidance for manipulators and mobile robots,” *The international journal of robotics research*, vol. 5, no. 1, pp. 90–98, 1986.
- [54] A. Howard, M. J. Mataric, and G. S. Sukhatme, “Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem,” *Distributed autonomous robotic systems*, vol. 5, pp. 299–308, 2002.
- [55] S. Alam and Z. J. Haas, “Coverage and connectivity in three-dimensional networks,” in *Proceedings of the 12th annual international conference on Mobile computing and networking*. ACM, 2006, pp. 346–357.
- [56] L. Liu, H. Ma, and X. Zhang, “On directional k-coverage analysis of randomly deployed camera sensor networks,” in *Communications, 2008. ICC’08. IEEE International Conference on*. IEEE, 2008, pp. 2707–2711.
- [57] A. Ghosh and S. K. Das, “Coverage and connectivity issues in wireless sensor networks: A survey,” *Pervasive and Mobile Computing*, 2008.
- [58] S. Ali, S. B. Qaisar, and E. A. Felemban, “A reliable connectivity based node placement strategy in linear and hierarchical wireless sensor networks,” in *2014 IEEE 25th Annual International Symposium on Personal, Indoor, and Mobile Radio Communication (PIMRC)*. IEEE, 2014, pp. 763–767.
- [59] C. Bettstetter, “On the minimum node degree and connectivity of a wireless multihop network,” in *Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing*. ACM, 2002, pp. 80–91.
- [60] W. Dargie and C. Poellabauer, *Fundamentals of wireless sensor networks: theory and practice*. John Wiley & Sons, 2010.
- [61] X. Li and Y. Ouyang, “Reliable sensor deployment for network traffic surveillance,” *Transportation research part B: methodological*, vol. 45, no. 1, pp. 218–231, 2011.
- [62] Z. Yun, X. Bai, D. Xuan, W. Jia, and W. Zhao, “Pattern mutation in wireless sensor deployment,” *IEEE/ACM Transactions on Networking*, vol. 20, no. 6, pp. 1964–1977, 2012.

- [63] M. Cardei, Y. Yang, and J. Wu, “Non-uniform sensor deployment in mobile wireless sensor networks,” in *World of Wireless, Mobile and Multimedia Networks, 2008. WoWMoM 2008. 2008 International Symposium on a.* IEEE, 2008, pp. 1–8.
- [64] N. Bartolini, T. Calamoneri, E. G. Fusco, A. Massini, and S. Silvestri, “Snap and spread: A self-deployment algorithm for mobile sensor networks,” in *International Conference on Distributed Computing in Sensor Systems.* Springer, 2008, pp. 451–456.
- [65] H. Jing, “Node deployment algorithm based on perception model of wireless sensor network,” *International Journal of Automation Technology*, vol. 9, no. 3, pp. 210–215, 2015.
- [66] T. Clouqueur, V. Phipatanasuphorn, P. Ramanathan, and K. K. Saluja, “Sensor deployment strategy for target detection,” in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications.* ACM, 2002, pp. 42–48.
- [67] J. Beutel, K. Römer, M. Ringwald, and M. Woehrle, “Deployment techniques for sensor networks,” in *Sensor Networks.* Springer, 2010, pp. 219–248.
- [68] V. Sharma, R. Patel, H. Bhadauria, and D. Prasad, “Deployment schemes in wireless sensor network to achieve blanket coverage in large-scale open area: A review,” *Egyptian Informatics Journal*, vol. 17, no. 1, pp. 45–56, 2016.
- [69] X. Liu and P. Mohapatra, “On the deployment of wireless sensor nodes,” in *Proceedings of the 3rd International Workshop on Measurement, Modeling, and Performance Analysis of Wireless Sensor Networks*, 2005.
- [70] A. Ababnah and B. Natarajan, “Optimal control-based strategy for sensor deployment,” *IEEE transactions on systems, man, and cybernetics-part a: systems and humans*, vol. 41, no. 1, pp. 97–104, 2011.
- [71] N. Akshay, M. P. Kumar, B. Harish, and S. Dhanorkar, “An efficient approach for sensor deployments in wireless sensor network,” in *Emerging Trends in Robotics and Communication Technologies (INTERACT), 2010 International Conference on.* IEEE, 2010, pp. 350–355.
- [72] M. T. Kouakou, S. Yamamoto, K. Yasumoto, and M. Ito, “Deployment planning tool for indoor 3d-wsns,” in *Proceedings of the 12th ACM international conference adjunct papers on Ubiquitous computing-Adjunct.* ACM, 2010, pp. 369–370.
- [73] C. W. Reynolds, “Flocks, herds and schools: A distributed behavioral model,” *ACM SIGGRAPH computer graphics*, vol. 21, no. 4, pp. 25–34, 1987.
- [74] C. Ozturk, D. Karaboga, and B. Gorkemli, “Probabilistic dynamic deployment of wireless sensor networks by artificial bee colony algorithm,” *Sensors*, vol. 11, no. 6, pp. 6056–6065, 2011.

- [75] S. Fidanova, P. Marinov, and E. Alba, "Ant algorithm for optimal sensor deployment," in *Computational Intelligence*. Springer, 2012, pp. 21–29.
- [76] J. C. Hou, D. K. Yau, C. Y. Ma, Y. Yang, H. Zhang, I.-H. Hou, N. S. Rao, and M. Shankar, "Coverage in wireless sensor networks," in *Guide to Wireless Sensor Networks*. Springer, 2009, pp. 47–79.
- [77] M. Cardei, M. T. Thai, Y. Li, and W. Wu, "Energy-efficient target coverage in wireless sensor networks," in *INFOCOM 2005. 24th annual joint conference of the IEEE computer and communications societies. proceedings IEEE*, vol. 3. IEEE, 2005, pp. 1976–1984.
- [78] X.-Y. Li, P.-J. Wan, and O. Frieder, "Coverage in wireless ad hoc sensor networks," *IEEE Transactions on computers*, vol. 52, no. 6, pp. 753–763, 2003.
- [79] S. Imran and Y.-B. Ko, "A continuous object boundary detection and tracking scheme for failure-prone sensor networks," *Sensors*, vol. 17, no. 2, p. 361, 2017.
- [80] A. Sangwan and R. P. Singh, "Survey on coverage problems in wireless sensor networks," *Wireless Personal Communications*, vol. 80, no. 4, pp. 1475–1500, 2015. [Online]. Available: <http://dx.doi.org/10.1007/s11277-014-2094-3>
- [81] B. Liu, O. Dousse, J. Wang, and A. Saipulla, "Strong barrier coverage of wireless sensor networks," in *Proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing*. ACM, 2008, pp. 411–420.
- [82] S. Yang, F. Dai, M. Cardei, J. Wu, and F. Patterson, "On connected multiple point coverage in wireless sensor networks," *International Journal of Wireless Information Networks*, vol. 13, no. 4, pp. 289–301, 2006.
- [83] J. Wang and N. Zhong, "Efficient point coverage in wireless sensor networks," *Journal of Combinatorial Optimization*, vol. 11, no. 3, pp. 291–304, 2006.
- [84] M. Esnaashari and M. R. Meybodi, "A learning automata based scheduling solution to the dynamic point coverage problem in wireless sensor networks," *Computer Networks*, vol. 54, no. 14, pp. 2410–2438, 2010.
- [85] S. Yang, F. Dai, M. Cardei, and J. Wu, "On multiple point coverage in wireless sensor networks," in *Mobile Adhoc and Sensor Systems Conference, 2005. IEEE International Conference on*. IEEE, 2005, pp. 8–pp.
- [86] H. Chen, H. Wu, and N.-F. Tzeng, "Grid-based approach for working node selection in wireless sensor networks," in *Communications, 2004 IEEE International Conference on*, vol. 6. IEEE, 2004, pp. 3673–3678.
- [87] X. Chen, Y.-C. Ho, and H. Bai, "Complete coverage and point coverage in randomly distributed sensor networks," *Automatica*, vol. 45, no. 6, pp. 1549–1553, 2009.
- [88] M. Chertok and A. Pinhas, "Increasing interest point coverage in an image," May 20 2014, uS Patent 8,731,306.

- [89] R. Katsuma, Y. Murata, N. Shibata, K. Yasumoto, and M. Ito, "Extending k-coverage lifetime of wireless sensor networks using mobile sensor nodes," in *Wireless and Mobile Computing, Networking and Communications, 2009. WIMOB 2009. IEEE International Conference on*. IEEE, 2009, pp. 48–54.
- [90] F. Al-Turjman, H. S. Hassanein, and M. A. Ibnkahla, "Connectivity optimization for wireless sensor networks applied to forest monitoring," in *Communications, 2009. ICC'09. IEEE International Conference on*. IEEE, 2009, pp. 1–6.
- [91] F. M. Al-Turjman, H. S. Hassanein, and M. A. Ibnkahla, "Connectivity optimization with realistic lifetime constraints for node placement in environmental monitoring," in *2009 IEEE 34th Conference on Local Computer Networks*, Oct 2009, pp. 617–624.
- [92] S. Sajadian, A. Ibrahim, E. P. de Freitas, and T. Larsson, "Improving connectivity of nodes in mobile wsn," in *Advanced Information Networking and Applications (AINA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 364–371.
- [93] B. Berkowitz and R. P. Ewing, "Percolation theory and network modeling applications in soil physics," *Surveys in Geophysics*, vol. 19, no. 1, pp. 23–72, 1998.
- [94] P. Wang, Z. Sun, M. C. Vuran, M. A. Al-Rodhaan, A. M. Al-Dhelaan, and I. F. Akyildiz, "On network connectivity of wireless sensor networks for sandstorm monitoring," *Computer Networks*, vol. 55, no. 5, pp. 1150–1157, 2011.
- [95] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer networks*, vol. 38, no. 4, pp. 393–422, 2002.
- [96] S. Megerian, F. Koushanfar, M. Potkonjak, and M. B. Srivastava, "Worst and best-case coverage in sensor networks," *Mobile Computing, IEEE Transactions on*, vol. 4, no. 1, pp. 84–92, 2005.
- [97] B. Tavli, K. Bicakci, R. Zilan, and J. M. Barcelo-Ordinas, "A survey of visual sensor network platforms," *Multimedia Tools and Applications*, vol. 60, no. 3, pp. 689–726, 2012.
- [98] K.-Y. Chow, K.-S. Lui, and E. Y. Lam, "Maximizing angle coverage in visual sensor networks," in *Communications, 2007. ICC'07. IEEE International Conference on*. IEEE, 2007, pp. 3516–3521.
- [99] H.-H. TYen, "Novel visual sensor deployment algorithm in ptz wireless visual sensor networks," in *Wireless and Mobile, 2014 IEEE Asia Pacific Conference on*. IEEE, 2014, pp. 214–218.
- [100] H.-H. Yen, "Efficient visual sensor coverage algorithm in wireless visual sensor networks," in *Wireless Communications and Mobile Computing Conference (IWCMC), 2013 9th International*. IEEE, 2013, pp. 1516–1521.

- [101] C.-K. Liang, C.-H. Tsai, and M.-C. He, "On area coverage problems in directional sensor networks," in *Information Networking (ICOIN), 2011 International Conference on*. IEEE, 2011, pp. 182–187.
- [102] J. Peng, J. Jingqi, W. Chengdong, and L. Hongchao, "A coverage-enhance scheduling algorithm for 3d directional sensor networks," in *Control and Decision Conference (CCDC), 2013 25th Chinese*. IEEE, 2013, pp. 2888–2892.
- [103] M.-C. Wu and W.-F. Lu, "On target coverage problem of angle rotatable directional sensor networks," in *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2013 Seventh International Conference on*. IEEE, 2013, pp. 605–610.
- [104] Y.-G. Fu, J. Zhou, and L. Deng, "Surveillance of a 2d plane area with 3d deployed cameras," *Sensors*, vol. 14, no. 2, pp. 1988–2011, 2014.
- [105] D. G. Costa, I. Silva, L. A. Guedes, F. Vasques, and P. Portugal, "Availability issues in wireless visual sensor networks," *Sensors*, vol. 14, no. 2, pp. 2795–2821, 2014.
- [106] S. M. B. Malek, M. M. Sadik, and A. Rahman, "On balanced k-coverage in visual sensor networks," *J. Netw. Comput. Appl.*, vol. 72, no. C, pp. 72–86, Sep. 2016. [Online]. Available: <http://dx.doi.org/10.1016/j.jnca.2016.06.011>
- [107] E. Yildiz, K. Akkaya, E. Sisikoglu, and M. Y. Sir, "Optimal camera placement for providing angular coverage in wireless video sensor networks," *IEEE transactions on computers*, vol. 63, no. 7, pp. 1812–1825, 2014.
- [108] M. Hefeeda and M. Bagheri, "Randomized k-coverage algorithms for dense sensor networks," in *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*. IEEE, 2007, pp. 2376–2380.
- [109] H. M. Ammari, "On the problem of k-coverage in 3d wireless sensor networks: A reuleaux tetrahedron-based approach," in *2011 Seventh International Conference on Intelligent Sensors, Sensor Net. and Info. Processing*, Dec 2011, pp. 389–394.
- [110] A. Raniwala and T.-c. Chiueh, "Architecture and algorithms for an ieee 802.11-based multi-channel wireless mesh network," in *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, vol. 3. IEEE, 2005, pp. 2223–2234.
- [111] Y. Charfi, N. Wakamiya, and M. Murata, "Challenging issues in visual sensor networks," *IEEE Wireless Communications*, vol. 16, no. 2, 2009.
- [112] W.-T. Wang and K.-F. Ssu, "Obstacle detection and estimation in wireless sensor networks," *Computer Networks*, vol. 57, no. 4, pp. 858–868, 2013.
- [113] R. Zheng, K. Vu, A. Pendharkar, and G. Song, "Obstacle discovery in distributed actuator and sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 7, no. 3, p. 22, 2010.

- [114] C.-Y. Chang, J.-P. Sheu, Y.-C. Chen, and S.-W. Chang, "An obstacle-free and power-efficient deployment algorithm for wireless sensor networks," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 39, no. 4, pp. 795–806, 2009.
- [115] H. Tan, Y. Wang, X. Hao, Q.-S. Hua, and F. C. Lau, "Arbitrary obstacles constrained full coverage in wireless sensor networks," in *International Conference on Wireless Algorithms, Systems, and Applications*. Springer, 2010, pp. 1–10.
- [116] D. G. Costa, I. Silva, L. A. Guedes, P. Portugal, and F. Vasques, "Enhancing redundancy in wireless visual sensor networks for target coverage," in *Proceedings of the 20th Brazilian Symposium on Multimedia and the Web*. ACM, 2014, pp. 31–38.
- [117] L. Guo, Y. Zhu, D. Li, and D. Kim, "Ptz camera scheduling for selected area coverage in visual sensor networks," in *Distributed Computing Systems (ICDCS), 2015 IEEE 35th International Conference on*. IEEE, 2015, pp. 379–388.
- [118] A. Neishaboori, A. Saeed, K. A. Harras, and A. Mohamed, "On target coverage in mobile visual sensor networks," in *Proceedings of the 12th ACM international symposium on Mobility management and wireless access*. ACM, 2014, pp. 39–46.
- [119] F. G. Yap and H.-H. Yen, "A survey on sensor coverage and visual data capturing/processing/transmission in wireless visual sensor networks," *Sensors*, vol. 14, no. 2, pp. 3506–3527, 2014.
- [120] V. P. Munishwar and N. B. Abu-Ghazaleh, "Coverage algorithms for visual sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 9, no. 4, p. 45, 2013.
- [121] N. Bendimerad and B. Kechar, "Rotational wireless video sensor networks with obstacle avoidance capability for improving disaster area coverage." *Journal of Information Processing Systems*, vol. 11, no. 4, 2015.
- [122] T. Puu and M. Beckmann, "Continuous space modelling," in *Handbook of transportation science*. Springer, 1999, pp. 269–310.
- [123] G. Birkhoff and S. Mac Lane, *A brief survey of modern algebra*. Macmillan, 1963.
- [124] D. G. Costa and L. A. Guedes, "Exploiting the sensing relevancies of source nodes for optimizations in visual sensor networks," *Multimedia tools and applications*, vol. 64, no. 3, pp. 549–579, 2013.
- [125] G. Xing, X. Wang, Y. Zhang, C. Lu, R. Pless, and C. Gill, "Integrated coverage and connectivity configuration for energy conservation in sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 1, no. 1, pp. 36–72, 2005.
- [126] Y. Wang, S. Lederer, and J. Gao, "Connectivity-based sensor network localization with incremental delaunay refinement method," in *INFOCOM 2009, IEEE*. IEEE, 2009, pp. 2401–2409.

- [127] S. Shakkottai, R. Srikant, and N. B. Shroff, “Unreliable sensor grids: Coverage, connectivity and diameter,” *Ad Hoc Networks*, vol. 3, no. 6, pp. 702–716, 2005.
- [128] G. Tan, S. A. Jarvis, and A.-M. Kermarrec, “Connectivity-guaranteed and obstacle-adaptive deployment schemes for mobile sensor networks,” *IEEE Transactions on Mobile Computing*, vol. 8, no. 6, pp. 836–848, 2009.
- [129] S. S. Dhillon and K. Chakrabarty, “Sensor placement for effective coverage and surveillance in distributed sensor networks,” in *Wireless Communications and Networking, 2003. WCNC 2003. 2003 IEEE*, vol. 3. IEEE, 2003, pp. 1609–1614.
- [130] O. Georgiou, C. P. Dettmann, and J. P. Coon, “Network connectivity: Stochastic vs. deterministic wireless channels,” in *Communications (ICC), 2014 IEEE International Conference on*. IEEE, 2014, pp. 77–82.
- [131] J. Muller, A. Rottmann, L. M. Reindl, and W. Burgard, “A probabilistic sonar sensor model for robust localization of a small-size blimp in indoor environments using a particle filter,” in *Robotics and Automation, 2009. ICRA’09. IEEE International Conference on*. IEEE, 2009, pp. 3589–3594.
- [132] N. Mehranbod, M. Soroush, M. Piovoso, and B. A. Ogunnaike, “Probabilistic model for sensor fault detection and identification,” *AIChE Journal*, vol. 49, no. 7, pp. 1787–1802, 2003.
- [133] A. El-Mougy, M. Ibnkahla, and L. Hegazy, “Software-defined wireless network architectures for the internet-of-things,” in *2015 IEEE 40th Local Computer Networks Conference Workshops (LCN Workshops)*, 2015, pp. 804–811.
- [134] S. S. Baidya and C. Bhattacharyya, “Coverage and connectivity in wireless sensor networks: Their trade-offs,” in *Sensing technology (ICST), 2012 sixth international conference on*. IEEE, 2012, pp. 353–358.
- [135] S. Sharmin, F. N. Nur, M. A. Razzaque, M. M. Rahman, A. Alelaiwi, M. M. Hassan, and S. M. M. Rahman, “ α -overlapping area coverage for clustered directional sensor networks,” *Computer Communications*, 2017.
- [136] C. Zhang, X. Bai, J. Teng, D. Xuan, and W. Jia, “Constructing low-connectivity and full-coverage three dimensional sensor networks,” *IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS*, vol. 28, no. 7, p. 1, 2010.
- [137] X. Deng, J. Yu, D. Yu, and C. Chen, “Transforming area coverage to target coverage to maintain coverage and connectivity for wireless sensor networks,” *International Journal of Distributed Sensor Networks*, vol. 8, no. 10, p. 254318, 2012. [Online]. Available: <http://dx.doi.org/10.1155/2012/254318>
- [138] A. Saeed, A. Abdelkader, M. Khan, A. Neishaboori, K. A. Harras, and A. Mohamed, “On realistic target coverage by autonomous drones,” *arXiv preprint arXiv:1702.03456*, 2017.

- [139] H. M. Ammari and S. K. Das, "Centralized and clustered k-coverage protocols for wireless sensor networks," *IEEE Transactions on Computers*, vol. 61, no. 1, pp. 118–133, 2012.
- [140] J. Yu, Y. Chen, L. Ma, B. Huang, and X. Cheng, "On connected target k-coverage in heterogeneous wireless sensor networks," *Sensors*, vol. 16, no. 1, p. 104, 2016.
- [141] Z. Zhou, S. Das, and H. Gupta, "Connected k-coverage problem in sensor networks," in *Proceedings. 13th International Conference on Computer Communications and Networks (IEEE Cat. No.04EX969)*, Oct 2004, pp. 373–378.
- [142] J. Carle and D. Simplot-Ryl, "Energy-efficient area monitoring for sensor networks," *Computer*, vol. 37, no. 2, pp. 40–46, Feb 2004.
- [143] D. G. Costa, L. A. Guedes, F. Vasques, and P. Portugal, "Research trends in wireless visual sensor networks when exploiting prioritization," *Sensors*, vol. 15, no. 1, pp. 1760–1784, 2015.
- [144] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm intelligence: from natural to artificial systems*. Oxford university press, 1999, no. 1.
- [145] D. Karaboga, B. Gorkemli, C. Ozturk, and N. Karaboga, "A comprehensive survey: artificial bee colony (abc) algorithm and applications," *Artificial Intelligence Review*, vol. 42, no. 1, pp. 21–57, 2014.
- [146] D. Karaboga and B. Akay, "A survey: algorithms simulating bee swarm intelligence," *Artificial intelligence review*, vol. 31, no. 1-4, pp. 61–85, 2009.
- [147] R. V. Kulkarni and G. K. Venayagamoorthy, "Particle swarm optimization in wireless-sensor networks: A brief survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 41, no. 2, pp. 262–267, 2011.
- [148] Q. Wang, "Traffic analysis & modeling in wireless sensor networks and their applications on network optimization and anomaly detection," *Network Protocols and Algorithms*, vol. 2, no. 1, pp. 74–92, 2010.
- [149] T. Brown, Z. Wang, T. Shan, F. Wang, and J. Xue, "On wireless video sensor network deployment for 3d indoor space coverage," in *SoutheastCon, 2016*. IEEE, 2016, pp. 1–8.
- [150] T. Brown *et al.*, "Poster abstract: Obstacle and connectivity aware wireless video sensor deployment for 3d indoor monitoring," in *2017 IEEE/ACM Second International Conference on Internet-of-Things Design and Implementation (IoTDI)*, April 2017, pp. 305–306.
- [151] Z. Wang, T. Brown, T. Shan, F. Wang, and J. Xue, "On 2-angular-coverage in wireless visual sensor network deployment for 3d indoor monitoring," in *Proceedings of the SouthEast Conference*. ACM, 2017, pp. 145–148.

LIST OF APPENDICES

APPENDIX A: GREEDY HEURISTIC ALGORITHM COMPLEXITY ANALYSIS

Appendix A

Greedy Heuristic Algorithm Complexity Analysis

The section below will discuss the implementation of the constraints modeled in the discrete version of our formulated problem for the greedy heuristic algorithm.

A greedy algorithm builds a solution in small incremental steps, selecting a decision to optimize some underlying criterion. In our case, the greedy heuristic algorithm identifies the locally optimal choice at each iteration phase of the algorithm, where it selects the video sensors that covers the maximum number of points that still need to be covered. This greedy strategy is utilized to yield a local optimal solution to approximate a global optimal solution in a feasible time. Moreover, this approach will provide significant benefits to our latter solutions by reducing the deployment candidate locations and additionally can provide a relative approximate solution to the optimization problem.

Herein, we are interested in the performance of the algorithm as the size of the input (i.e., n) becomes large, which is generally determined by the number of deployment locations for the video sensors within the monitored area. The algorithm also sorts the deployment location based on the number of fresh points that will be covered if placed at that locations. In both algorithms, we use quick-sort to sort the locations. The worst case for a quick sort algorithm is n^2 . Given these factors, the number of operations required by our algorithm is $\mathbf{O}(n^3)$, (i.e., there are n iterations for the input size times each iteration n^2 for quick sorting for worst case and with quick sorting on average, it may be $\mathbf{O}(n^2 \log n)$). However, if in each iteration instead of sorting the candidate locations, we can directly select the video sensor that covers the maximum number of fresh points, then the overall complexity can be reduced to $\mathbf{O}(n^2)$.

APPENDIX B: ENHANCED DFS ALGORITHM COMPLEXITY ANALYSIS

Appendix B

Enhanced DFS Algorithm Complexity Analysis

The section below will discuss the implementation of the constraints modeled in the discrete version of our formulated problem for our enhanced depth first search algorithm.

In the enhanced depth first search algorithm, we implement some enhancements to improve the greedy solution. We start the search with the current best found solution list from the greedy search approach discussed above. The strategy assist the enhanced DFS algorithm in removing the search branches that have already used equal or more number of video sensors compared to the currently found best solution and can safely prune them. We then sort the candidate locations for video sensor deployment based on the maximum number of fresh point particles that it can cover. With respect to the sensor angle direction, we derive a lower bound. Finally, we return a list with location and direction for camera sensor. Traditional depth first search algorithms often do not utilize a pruning technique.

Traditional depth first search algorithm are used to traverse an entire search graph, where the time complexity of the algorithm is $\mathbf{O}(V + E)$, (i.e., worst case where V denotes the vertices and E denotes the edges within the graph). For the implementation of our enhanced DFS as it relates to the discrete 3D grid, the search is performed on a reduced solution space set (i.e., currently found best solution list from the greedy heuristic algorithm). Furthermore, we use a pruning technique to limit the depth space that must be explored. Given, these factors, in the worst case the time complexity of the algorithm can reach $\mathbf{O}(n!)$, considering the candidate location and facing direction. However, on average our algorithm is not near the worst case scenario. For example, to start the search with the current best found solution list from the greedy search approach and use pruning, the complexity can be

reduced to $\mathbf{O}(\prod_{i=n-n'+1}^n i)$, where n' is the number of required video sensors found by the greedy search approach.

APPENDIX C: OBSTACLE CONSTRAINT CODE LISTING

Appendix C

Code Listing: Obstacle Constraint

```
/* Sample coding of the obstacle-awareness constraint */

/* Variables to iterate through the particles in space */

int innerCounter = 0;
for(int pointIndex=0;pointIndex<cloudPoints.size();pointIndex++) {

/* Boolean parameter for obstacle detection */

    boolean obstacleCheck = false;

/* Variables to hold video sensor location */
    Vector3 localTemp = cameras.get(camIndex).cpy();

/* Iterate through all obstacles*/

    for(int ii=0;ii<obstacle.size();ii++){
        if(obstacleCheck) {
            break;

        } else {

float t = 0;
float tempX = 0;
float tempY = 0;
float tempZ = 0;

/* Check if points are obstructed by the face (i.e., obstacle planes) of the obstacle */

if((localTemp.x<=obstacle.get(ii)[0]&&obstacle.get(ii)[0]<=cloudPoints.get(pointIndex).x)||
    localTemp.x>=obstacle.get(ii)[0]&&obstacle.get(ii)[0]>=cloudPoints.get(pointIndex).x)) {
```



```

t = (obstacle.get(ii)[0] - localTemp.x) / (cloudPoints.get(pointIndex).x - localTemp.x);
tempY = t * (cloudPoints.get(pointIndex).y - localTemp.y) + localTemp.y;
tempZ = t * (cloudPoints.get(pointIndex).z - localTemp.z) + localTemp.z;

if (tempY >= obstacle.get(ii)[4] && tempY <= obstacle.get(ii)[5] && tempZ >= obstacle.get(
ii)[2] && tempZ <= obstacle.get(ii)[3]) {
    obstacleCheck = true;
    break;
}

if ((localTemp.x <= obstacle.get(ii)[1] && obstacle.get(ii)[1] <= cloudPoints.get(pointIndex).x) || (
localTemp.x >= obstacle.get(ii)[1] && obstacle.get(ii)[1] >= cloudPoints.get(pointIndex).x)) {
    t = (obstacle.get(ii)[1] - localTemp.x) / (cloudPoints.get(pointIndex).x - localTemp.x);
    tempY = t * (cloudPoints.get(pointIndex).y - localTemp.y) + localTemp.y;
    tempZ = t * (cloudPoints.get(pointIndex).z - localTemp.z) + localTemp.z;

    if (tempY >= obstacle.get(ii)[4] && tempY <= obstacle.get(ii)[5] && tempZ >= obstacle.get(
ii)[2] && tempZ <= obstacle.get(ii)[3]) {
        obstacleCheck = true;
        break;
    }

    if ((localTemp.z <= obstacle.get(ii)[2] && obstacle.get(ii)[2] <= cloudPoints.get(pointIndex).z) || (
localTemp.z >= obstacle.get(ii)[2] && obstacle.get(ii)[2] >= cloudPoints.get(pointIndex).z)) {
        t = (obstacle.get(ii)[2] - localTemp.z) / (cloudPoints.get(pointIndex).z - localTemp.z);
        tempY = t * (cloudPoints.get(pointIndex).y - localTemp.y) + localTemp.y;
        tempX = t * (cloudPoints.get(pointIndex).x - localTemp.x) + localTemp.x;

        if (tempY >= obstacle.get(ii)[4] && tempY <= obstacle.get(ii)[5] && tempX >= obstacle.get(
ii)[0] && tempX <= obstacle.get(ii)[1]) {
            obstacleCheck = true;
            break;
        }

        if ((localTemp.z <= obstacle.get(ii)[3] && obstacle.get(ii)[3] <= cloudPoints.get(pointIndex).z) || (
localTemp.z >= obstacle.get(ii)[3] && obstacle.get(ii)[3] >= cloudPoints.get(pointIndex).z)) {
            t = (obstacle.get(ii)[3] - localTemp.z) / (cloudPoints.get(pointIndex).z - localTemp.z);

            tempY = t * (cloudPoints.get(pointIndex).y - localTemp.y) + localTemp.y;
            tempX = t * (cloudPoints.get(pointIndex).x - localTemp.x) + localTemp.x;

```

```

    if (tempY >= obstacle.get(ii)[4] && tempY <= obstacle.get(ii)[5] && tempX >= obstacle.
get(ii)[0] && tempX <= obstacle.get(ii)[1]) {
        obstacleCheck = true;
        break;
    }
}

if((localTemp.y<=obstacle.get(ii)[4]&&obstacle.get(ii)[4]<=cloudPoints.get(pointIndex).y)||
    localTemp.y>=obstacle.get(ii)[4]&&obstacle.get(ii)[4]>=cloudPoints.get(pointIndex).y)) {
    t = (obstacle.get(ii)[4] - localTemp.y) / (cloudPoints.get(pointIndex).y - localTemp.y);
    tempZ = t * (cloudPoints.get(pointIndex).z - localTemp.z) + localTemp.z;
    tempX = t * (cloudPoints.get(pointIndex).x - localTemp.x) + localTemp.x;

    if (tempZ >= obstacle.get(ii)[2] && tempZ <= obstacle.get(ii)[3] && tempX >= obstacle.get(
        ii)[0] && tempX <= obstacle.get(ii)[1]) {
        obstacleCheck = true;
        break;
    }
}

if((localTemp.y<=obstacle.get(ii)[5]&&obstacle.get(ii)[5]<=cloudPoints.get(pointIndex).y)||
    localTemp.y>=obstacle.get(ii)[5]&&obstacle.get(ii)[5]>=cloudPoints.get(pointIndex).y)) {
    t = (obstacle.get(ii)[5] - localTemp.y) / (cloudPoints.get(pointIndex).y - localTemp.y);
    tempZ = t * (cloudPoints.get(pointIndex).z - localTemp.z) + localTemp.z;
    tempX = t * (cloudPoints.get(pointIndex).x - localTemp.x) + localTemp.x;

    if (tempZ >= obstacle.get(ii)[2] && tempZ <= obstacle.get(ii)[3] && tempX >= obstacle.get(
        ii)[0] && tempX <= obstacle.get(ii)[1]) {
        obstacleCheck = true;
        break;
    }
}
}
}
}

```

APPENDIX D: CONNECTIVITY CONSTRAINT CODE LISTING

Appendix D

Code Listing: Connectivity Constraint

```
/* Sample coding of the connectivity constraint */

/* Variables to check video sensor communication range */
boolean connectivityCheckFlag = false;
double rangeCheckFlag = 0;

/* Iterate through video sensor list*/
for (int finalI =0; finalI<finalList.size();finalI++) {

/* Calculate distance between video sensor nodes */

rangeCheckFlag = Math.sqrt(Math.pow(Math.abs(finalList.get(finalI).position.x-
    candidateLocations.get(i).x),2)+Math.sqrt(Math.pow(Math.abs(finalList.get(finalI).
    position.x-candidateLocations.get(i).y),2)+Math.sqrt(Math.pow(Math.abs(finalList.get(
    finalI).position.x-candidateLocations.get(i).z),2));

/* Check if video sensor is within communication range */

if (rangeCheckFlag<=100){
    connectivityCheckFlag=true;
    break;
}
}

/* Continue if video sensor is within range */

if(!connectivityCheckFlag){

continue;
}

}
```

APPENDIX E: ANGULAR COVERAGE CONSTRAINT CODE LISTING

Appendix E

Code Listing: Angular Coverage Constraint

```
/* Sample code of the angular-coverage constraint */

/* Check if particles are covered by any video sensor */

if (particles.get(n).coverCount==0) {

particles.get(n).cameral=camera.position;
particles.get(n).coverCount++;

/* Particles are covered by at least one video sensor (i.e., 1-coverage), next test to see
   if they can achieve 2-coverage */

} else if(particles.get(n).coverCount==1) {

double cosAngle=particles.get(n).cameral.x-particles.get(n).particle.x)*(particles.get(n).
particles.x-camera.position.x)+particles.get(n).camera.ly-particles.get(n).particle.y)*(
particles.get(n).particles.y-camera.position.y)+particles.get(n).cameral.z-particles.get
(n).particle.z)*(particles.get(n).particles.z-camera.position.z))/(Math.sqrt(Math.pow((
particles.get(n).cameral.x-particles.get(n).particle.x),2)+Math.pow((particles.get(n).
cameral.y-particles.get(n).particle.y),2)+Math.pow((particles.get(n).cameral.z-particles
.get(n).particle.z),2))*(Math.sqrt(Math.pow((particles.get(n).particle.x-camera.position
.x),2)+Math.pow((particles.get(n).particle.y-camera.position.y),2)+Math.pow((particles
.get(n).particle.z-camera.position.z),2)));

/* Variable to check video sensor communication range is greater or equal to 120 ^\circ */

if(cosAngle<=-0.5) {
innerCounter++;
}

}
```

VITA

Education

2017: PH.D. in Computer Science	UNIVERSITY OF MISSISSIPPI
2009: M.S. in Computer Science	JACKSON STATE UNIVERSITY
2006: B.S. in Computer Science	MISSISSIPPI VALLEY STATE UNIVERSITY

Professional Experience

July - Aug 2017: John Hopkins University CTY Instructor, Easton, PA
Aug. '14 - May 2017: Turning Technologies Intern, Youngstown, OH
May - Aug. 2013: Mickey Leland Energy Fellowship Intern, Morgantown, WV
May - Aug. 2010: Space Grant Fellow Fellowship Intern, Pasadena, CA
May - Aug. 2008: NGI / NOAA Fellow Intern, Jackson, MS
May - Aug. 2007: National Data Buoy Center Intern, Picayune, MS
May - Aug. 2006: Radiance Technologies Intern, Jackson, MS
May - Aug. 2005: McNair Scholars Program Fellow Intern, Akron, OH
May - Aug. 2004: High Performance Computing JSU/UAB Intern, Jackson, MS

Publications

1. Z. Wang, T. Brown, T. Shan, F. Wang, and J. Xue. (2017), "On 2-Angular-Coverage in Wireless Visual Sensor Network Deployment for 3D Indoor Monitoring", *In Proceedings of the SouthEast Conference (ACM SE '17)*. ACM, New York, NY, USA, 145-148. DOI: <https://doi.org/10.1145/3077286.3078713>

2. Brown, T., Wang, Z., Shan, T., Wang, F., AND Xue, J., (2016), “On Wireless Video Sensor Network Deployment for 3D Indoor Space Coverage”, *In Proceedings of the Spring 2016 IEEE SouthEastCon* Norfolk, VA, March 30 - April 3, 2016. pp. 1-8., doi: 10.1109/SECON.2016.7506744
3. Abu, A., Humos, E., Brown, T., Watts, M., Price, K. (2011), “A Web-Based Controlled System for Autonomous Agent”, *In Proceedings of the 2011 World Congress in Computer Science, Computer Engineering and Applied Computing (ICAI)* Las Vegas, Nevada, July, 2011., pp. 1-4., doi=10.1.1.217.8969
4. Brown, T., Wang, Z., Shan, T., Wang, F., AND Xue, J., “Obstacle-Aware Wireless Visual Sensor Network Deployment for 3D Indoor Monitoring”, Submitted to *IEEE GLOBECOM '17*.
5. Brown, T., Wang, Z., Shan, T., Wang, F., AND Xue, J., ‘Achieving 2-Angular-Coverage in 3D Wireless Visual Sensor Networks for Internet-of-Things based Indoor Monitoring’, To be submitted to *Peer-to-Peer Networking and Applications Journal Special Issue*, August 2017.

Posters

6. T. Brown, Z. Wang, T. Shan, F. Wang, and J. Xue. (2017), “Obstacle and Connectivity Aware Wireless Video Sensor Deployment for 3D Indoor Monitoring: Poster Abstract”. *In Proceedings of the Second International Conference on Internet-of-Things Design and Implementation (IoTDI '17)*. ACM, New York, NY, USA, 305-306. DOI: <https://doi.org/10.1145/3054977.3057924>
7. T. Brown, Z. Wang, T. Shan, F. Wang, and J. Xue. (2016)., “Obstacle Aware Wireless Visual Sensor Network Deployment for 3D Indoor Monitoring”, *In Spring 2016 BWAC I/UCRC IAB* Tucson, AZ, March 28-30, 2016.

8. T. Brown, Z. Wang, T. Shan, F. Wang, and J. Xue. (2015)., ‘On Wireless Video Sensor Network Deployment for 3D Indoor Space Coverage, In Fall 2015 BWAC I/UCRC IAB Oxford,MS, November 3-4, 2015.
9. Brown, T., A. E. Humos, A Web Based Controlled System for a Wireless Autonomous Agent, *In Proceedings of the 8TH Spring International Conference on Computing, Communications and Control Technologies (CCCT)*, Orlando, FL, April, 2010, PP 7-10.

Scholarship and Awards

2017: DOCTORAL DISSERTATION FELLOWSHIP, UNIVERSITY OF MISSISSIPPI

2014 - 2017: GRADUATE MINORITY FELLOWSHIP, UNIVERSITY OF MISSISSIPPI

2014 - 2017: GRADUATE ASST. SCHOLARSHIP, UNIVERSITY OF MISSISSIPPI

2011 - 2014: SREB FELLOWSHIP, UNIVERSITY OF MISSISSIPPI

2009 - 2011: MSSGC FELLOWSHIP, UNIVERSITY OF MISSISSIPPI

Travel Grants

2017 GSC 7th Annual Research Symposium (300 USD), Uni. of Mississippi

2016 Graduate Travel Award, IEEE Southeast CON’16 (1K USD), Uni. of Mississippi

Leadership/Involvement

2014 - 2016 Graduate Student Council (GSC) Senator, University of Mississippi

2010 - 2012 GSC: Director of Technology, University of Mississippi

2009 - 2011 Oxford Mentorship Program , University of Mississippi

2009 Black Student Union, University of Mississippi

Service to the profession

- 2017 - ACM Southeast Conference, External Reviewer (The Annual ACM Southeast Conference Featuring Multidisciplinary and Interdisciplinary Computing) —Southeast

Conference '17 Track.

- 2016 - IEEE SoutheastCon, External Reviewer (The IEEE Region 3 and the IEEE Hampton Roads Section)—SoutheastCon '16 Track.
- 2016 - IEEE SoutheastCon, Student Volunteer (The IEEE Region 3 and the IEEE Hampton Roads Section)—SoutheastCon '16 Track.
- 2015 - Broadband Wireless Access Applications Workshop (BWAC I/UCRC IAB) Student Volunteer, Fall'15.
- 2015 - Broadband Wireless Access Applications Workshop (BWAC I/UCRC IAB) Student Volunteer, Spring'15.